

Avances en Inteligencia Artificial

Research in Computing Science

Series Editorial Board

Comité Editorial de la Serie

Editors-in-Chief:

Editores en Jefe

Juan Humberto Sossa Azuela (Mexico)

Gerhard Ritter (USA)

Jean Serra (France)

Ulises Cortés (Spain)

Associate Editors:

Editores Asociados

Jesús Angulo (France)

Jihad El-Sana (Israel)

Jesús Figueroa (Mexico)

Alexander Gelbukh (Russia)

Ioannis Kakadiaris (USA)

Serguei Levachkine (Russia)

Petros Maragos (Greece)

Julian Padget (UK)

Mateo Valero (Spain)

Editorial Coordination:

Coordinación Editorial

Blanca Miranda Valencia

Research in Computing Science es una publicación trimestral, de circulación internacional, editada por el Centro de Investigación en Computación del IPN, para dar a conocer los avances de investigación científica y desarrollo tecnológico de la comunidad científica internacional. **Volume 55**, junio, 2012. Tiraje: 500 ejemplares. *Certificado de Reserva de Derechos al Uso Exclusivo del Título* No. : 04-2005-121511550100-102, expedido por el Instituto Nacional de Derecho de Autor. *Certificado de Licitud de Título* No. 12897, *Certificado de licitud de Contenido* No. 10470, expedidos por la Comisión Calificadora de Publicaciones y Revistas Ilustradas. El contenido de los artículos es responsabilidad exclusiva de sus respectivos autores. Queda prohibida la reproducción total o parcial, por cualquier medio, sin el permiso expreso del editor, excepto para uso personal o de estudio haciendo cita explícita en la primera página de cada documento. Impreso en la Ciudad de México, en los Talleres Gráficos del IPN – Dirección de Publicaciones, Tres Guerras 27, Centro Histórico, México, D.F. Distribuida por el Centro de Investigación en Computación, Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othón de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738, México, D.F. Tel. 57 29 60 00, ext. 56571.

Editor Responsible: *Juan Humberto Sossa Azuela, RFC SOAJ560723*

Research in Computing Science is published by the Center for Computing Research of IPN. **Volume 55**, June, 2012. Printing 500. The authors are responsible for the contents of their articles. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without prior permission of Centre for Computing Research. Printed in Mexico City, November, 2011, in the IPN Graphic Workshop – Publication Office.

Volume 55

Volumen 55

Avances en Inteligencia Artificial

Editor del volumen:

Volume Editor

Grigori Sidorov

Instituto Politécnico Nacional
Centro de Investigación en Computación
México 2012



ISSN: 1870-4069

Copyright © Instituto Politécnico Nacional 2012

Copyright © by *Instituto Politécnico Nacional*

Instituto Politécnico Nacional (IPN)
Centro de Investigación en Computación (CIC)
Av. Juan de Dios Bátiz s/n esq. M. Othón de Mendizábal
Unidad Profesional “Adolfo López Mateos”, Zacatenco
07738, México D.F., México

<http://www.ipn.mx>

<http://www.cic.ipn.mx>

The editors and the Publisher of this journal have made their best effort in preparing this special issue, but make no warranty of any kind, expressed or implied, with regard to the information contained in this volume.

All rights reserved. No part of this publication may be reproduced, stored on a retrieval system or transmitted, in any form or by any means, including electronic, mechanical, photocopying, recording, or otherwise, without prior permission of the Instituto Politécnico Nacional, except for personal or classroom use provided that copies bear the full citation notice provided on the first page of each paper.

Indexed in LATINDEX and Periodica / Indexada en LATINDEX y Periódica

Printing: 500 / Tiraje: 500

Printed in Mexico / Impreso en México

Preface

El propósito de este volumen es reflejar las nuevas direcciones de investigación en el área de inteligencia artificial. Los artículos de este volumen fueron seleccionados a base de la revisión efectuada por los miembros del Comité editorial.

Este volumen contiene 35 artículos relacionados con varios aspectos de la inteligencia artificial. Los artículos se dividen en las siguientes 9 áreas temáticas:

- Procesamiento de lenguaje natural (6 artículos).
- Algoritmos bioinspirados (2 artículos).
- Ontologías, gestión y representación de conocimiento (6 artículos).
- Agentes (3 artículos).
- Lógica y razonamiento (2 artículos).
- Algoritmos y aplicaciones (6 artículos).
- Seguridad informática (2 artículos).
- Procesamiento de imágenes (5 artículos).
- Sistemas inteligentes de tutoría y aprendizaje (3 artículos).

Este volumen de la revista puede ser interesante para los investigadores y estudiantes de las ciencias de la computación, especialmente en áreas relacionadas con la inteligencia artificial; mismo que para el público en general interesado en este fascinante tema.

Expresamos nuestro agradecimiento al M.A. Pedro Ramírez Legorreta, Encargado de Rectoría de la Universidad Tecnológica de Xicotepec de Juárez, así mismo, al Comité Local del Congreso COMIA 2012 representado por la Academia de Tecnologías de la Información de la U.T.X.J., quien impulsa a nuevos horizontes de la ciencia a la comunidad universitaria de la Sierra Norte del Estado de Puebla.

El proceso de selección de división de artículos se llevó a cabo usando el sistema libremente disponible EasyChair, www.EasyChair.org.

Junio, 2012

Grigori Sidorov

Table of Contents

Índice

Page/Pág.

Procesamiento de lenguaje natural

Razonamiento analógico: una herramienta en la creación de narrativas	3
<i>Edgar Morales-Palafox y Rafael Pérez-y-Pérez</i>	
Extracción automática de hechos en libros de texto basada en estructuras sintácticas.....	15
<i>Honorato Aguilar-Galicia, Grigori Sidorov y Yulia Ledeneva</i>	
Agrupación de documentos utilizando representaciones holográficas reducidas	27
<i>Norma L. Cuautle Rivera, Maya Carrillo y Aurelio López-López</i>	
Evaluación de características para el proceso de atribución de autoría	37
<i>Esteban Castillo, Darnes Vilariño, David Pinto, Maya Carrillo e Iván Olmos</i>	
Análisis de una aplicación multilingüe del agrupamiento de textos	45
<i>Alisa Zhila y Alexander Gelbukh</i>	
Análisis del conjunto de datos PISA 2009 mediante técnicas de agrupamiento.....	57
<i>Yuridiana Alemán, David Pinto, Nahun Loya y Helena Gómez Adorno</i>	

Algoritmos bioinspirados

Distribución de puntos de red inalámbricos utilizando algoritmos genéticos	69
<i>M. Angélica Pinninghoff J., Ricardo Contreras A. y Rodrigo Aravena A.</i>	
Algoritmo alternativo de búsqueda tabú para la solución del problema de diseño económico estadístico de diagramas de control $\bar{X}, \bar{X} - S$	81
<i>Santiago Omar Caballero Morales y Felipe Trujillo Romero</i>	

Ontologías, gestión y representación de conocimiento

Introducción a las tecnologías de la Web semántica.....	93
<i>Alfredo Villagrán-Olguín, Neil Hernández-Gress y Miguel González Mendoza</i>	
Geontomex: una ontología espacial de México para la desambiguación de topónimos	103
<i>Belém Priego Sánchez, María J. Somodevilla García, Ivo H. Pineda Torres y Jaime Hernández Gómez</i>	

Base de Conocimiento de los procesos de desarrollo de software a través de un modelo de un sistema de gestión del conocimiento	113
<i>Sergio Ruiz-Castilla, Yulia Ledeneva y Rocío Morales</i>	
Modelo para diseño de perfiles de conocimiento: una aplicación en la industria generadora de energía eléctrica	125
<i>María de Jesús Velázquez Mendoza, Oscar Mario Rodríguez-Elías, César Enrique Rose Gómez y Sonia Regina Meneses Mendoza</i>	
Generación semiautomática de ontologías utilizando bases de datos relacionales como fuente primaria de información	137
<i>David González-Marrón, Miguel González-Mendoza y Neil Hernández-Gress</i>	
Sistema basado en conocimiento para la industria del vestido en México	151
<i>Karina Santiago Santiago, Ana Lilia Laureano Cruces, Jorge Sanchez De Antuñano, Estela Sarmiento Bustos y Oscar Domínguez Pérez</i>	

Agentes

Tres modelos computacionales de filosofía política	165
<i>José Martín Castro-Manzano</i>	
Manejo de la cadena de suministro mediante un sistema con múltiples agentes colaborativos	177
<i>Carlos R. Jaimez-González</i>	
Monitoreo de variables críticas en procesos industriales mediante una arquitectura multiagente	187
<i>Luis Arturo Madrid Hurtado, Oscar Mario Rodríguez-Elías, Germán Alonso Ruiz Domínguez y Guillermo Valencia Palomo</i>	

Lógica y razonamiento

Complejidad computacional de un caso especial del cálculo de grupos en las redes de interacción.....	199
<i>Roberto Alonso, Raúl Monroy y Luis A. Trejo</i>	
Lenguaje P-log: estructura y traducción a SMOBELS	209
<i>Daniel Rodríguez, Claudia Zepeda y José Luis Carballido</i>	

Algoritmos y aplicaciones

Implementación de una hiperheurística para la reducción de conflictos en el bloque quirúrgico	221
<i>Marina I. Ramos-Martínez, Martín Carpio, Martín Soria-Alcaraz y Héctor José Puga-Soberanes</i>	

Modelado con selección de estructura automática para sistemas no lineales usando SVM difusos	231
<i>Julio César Tovar Rodríguez</i>	
Mejorando el clasificador asociativo de patrones en bases de datos multi-clase	239
<i>Roberto O. Cruz L., Raúl Santiago M. y Héctor J. Puga S.</i>	
Análisis probabilístico mediante redes bayesianas para el cálculo de la probabilidad de lluvia en diferentes localidades	249
<i>Guillermo De la Torre-Gea y Oscar Delfín-Santiesteban</i>	
Modelado del problema de reparto utilizando un enfoque de planificación basado en Answer Set Programming	259
<i>Adriana Huitzil Tello, Claudia Zepeda Cortés y Mauricio Osorio Galindo</i>	
Implementación de la estrategia de juego "Tic-Tac-Toe" para la interacción con un brazo robótico	271
<i>Dora-Luz Almanza-Ojeda, Francisco Luna-Rodríguez, David Isaac Hernández Santos, Esteban Pérez Flores y Mario-Alberto Ibarra-Manzano</i>	

Seguridad informática

Hacia la detección de impostores mediante grafos de navegación	285
<i>J. Benito Camiña, Carlos Hernández-Gracidas y Raúl Monroy</i>	
Un nuevo esquema para la detección de ataques en redes inalámbricas	297
<i>Jorge Vázquez, Raúl Monroy y Luis A. Trejo</i>	

Procesamiento de imágenes

Orden de barrido optimizado para el cancelamiento de imágenes intra dañadas	311
<i>Kenia Picos-Espinoza, Mireya García-Vázquez y Alejandro A. Ramírez-Acosta</i>	
Propuesta de un algoritmo híbrido para la corrección de imágenes RAPD	321
<i>Ricardo Contreras A., M. Angélica Pinnighoff J. y Alejandra Alvarado V.</i>	
Descripción del borde utilizando medidas de compacidad aplicado al diagnóstico automático de cáncer de piel	333
<i>Raul Santiago-Montero, Noemí Barajas y J.L. Fernando Suárez Y Gómez</i>	
Asimetría de regiones digitales bajo el concepto de compacidad en el diagnóstico de cáncer de piel	343
<i>Raul Santiago, Antonio Zamarrón y Angélica Ortiz-Gaucin</i>	

Algoritmos de procesamiento del iris para un sistema de reconocimiento biométrico	353
<i>Ulises Orozco-Rosas, Mireya García-Vazquez y Alejandro A. Ramírez-Acosta</i>	

Sistemas inteligentes de tutoría y aprendizaje

Sistema tutor inteligente para la enseñanza de la programación orientada a objetos con el lenguaje C#.....	365
<i>Roberto J. Estrada Benavides, Laura Villavicencio Gómez y Boris A. Aranda Benítez</i>	
Un tutor inteligente y afectivo dentro de una red social para aprendizaje de matemáticas	377
<i>María L. Barrón-Estrada, Ramón Zatarain-Cabada, J.M. de Jesús López Arizmendi, José Luis Olivares Camacho y Karen Yadira González Trapero</i>	
Sistema de red inteligente Mindmapping (SRIM)	387
<i>Rosa María Rodríguez Aguilar, Sabino Miranda-Jiménez y Doricela Gutiérrez Cruz</i>	

Editorial Board of the Volume	393
Comité editorial del volumen	

Author Index	395
Índice de autores	

Procesamiento de lenguaje natural

Razonamiento analógico: una herramienta en la creación de narrativas

Edgar Morales-Palafox¹ y Rafael Pérez-y-Pérez²

¹ Posgrado en Ciencias e Ingeniería de la Computación, UNAM, D.F., México
edgar_morales_p@comunidad.unam.mx

² Departamento de Tecnologías de la Información,
División de Ciencias de la Comunicación y Diseño,
UAM Unidad Cuajimalpa, D.F., México
rperez@correo.cua.uam.mx

Resumen El presente trabajo propone un modelo computacional basado en el razonamiento analógico aplicado a un agente computacional generador de historias, con la finalidad de darle una nueva alternativa que permita saber cómo continuar la historia en un momento específico, cuando el conocimiento con el que cuenta no es suficiente para determinar el rumbo que debe seguir la narrativa. El razonamiento analógico se elige porque permite comprender problemas poco conocidos, a través de relaciones de similitud con experiencias almacenadas en nuestra memoria. La evaluación del modelo se hace en dos fases: primero se verifica que el modelo resuelva los bloqueos, resultando con una mejoría de 60%. En la segunda se valida que las historias sigan siendo coherentes e interesantes, con la intervención de la analogía. El modelo resulta benéfico si se utiliza cuando el bloqueo surge al inicio o al final de la creación de la narrativa.

Palabras clave: Razonamiento analógico, analogía, procesos cognitivos, cómputo creativo, creatividad, historias.

1. Introducción

Mexica (Pérez, 1999; 2007) escribe argumentos de historias cortas acerca de los antiguos Mexicanos. Algunas veces, cuando el programa intenta generar un nuevo argumento, se presentan situaciones en las que el sistema no sabe cómo continuar la historia; a esto se le denomina *impasse*. Mexica cuenta actualmente con un método propio (MP) para resolver el problema; sin embargo, algunas veces el MP no encuentra una solución, resultando en una historia que termina abruptamente generando una historia inconclusa. Por consiguiente, se pretende dotar al agente escritor con una alternativa más y con ello reducir el número de *impasses* no resueltos.

De acuerdo con la problemática que se mencionó anteriormente, se buscan alternativas, encontrando que el razonamiento analógico es una de las más adecuadas, por dos razones:

1. La ciencia cognoscitiva ha realizado algunas investigaciones en las que se proponen modelos computacionales que emplean el razonamiento analógico (RA) como

una forma de enfrentar los problemas que son poco conocidos. Algunos de estos modelos se presentan a continuación:

- COPYCAT (Hofstadter & Mitchell, 1995) se enfoca en descubrir analogías, en un contexto compuesto por series de letras del alfabeto, de la “A” a la “Z”.
- ACME (Holyoak & Thagard, 1989) el tipo de analogías que realiza son: Efecto de nivelación de solventes y abrir frascos, Enlaces químicos y estira y afloja, Distribución de las velocidades moleculares y distribución de la riqueza, entre otras.
- ACT-R (Salvucci & Anderson, 2001) es una propuesta de arquitectura cognoscitiva: una teoría acerca de cómo funciona los procesos del pensamiento humano, los investigadores realizaron pruebas del razonamiento analógico que se produce entre el sistema solar y el modelo atómico de Rutherford.

Cabe mencionar que los modelos RA no han sido utilizados como parte de la generación de narrativas, lo que hace atractivo este proyecto.

La segunda razón es porque autores como Mitchell (1993), Koestler (1964), Perkins (1981), Hofstadter (1995), Holyoak y Hummel (2002), Dartnall(2002), sugieren la existencia de una relación entre los procesos creativos y la analogía.

Como base las investigaciones anteriores, el objetivo del presente trabajo es crear un modelo computacional que utilice la analogía como una alternativa que permita minimizar el problema del *impasse*. Para cumplir con dicho objetivo, el presente trabajo se dividió en dos fases: en la primera, se trata de verificar que el modelo de RA cumpla con el propósito de minimizar el número de *impasses* que provocan historias inconclusas; en la segunda, dado que el modelo considera proponer una acción para continuar la historia interviniendo en el desarrollo del relato, se evalúa que la narrativa siga manteniendo su alto nivel de coherencia e interés. Para evaluar las historias en el presente trabajo se consideraron las siguientes variables:

- Fluidez: la secuencia de acciones se lleva de una forma clara, entendible para el lector.
- Estructura: el orden y la combinación de las acciones es adecuada.
- Contenido: se verifica el tipo de acciones que son utilizadas en la narrativa.
- Suspense: la variable que indica el grado de expectativa que se genera en el lector.
- Calidad: se verifica el cumplimiento de los criterios establecidos: fluidez, estructura, contenido y suspense.
- Gusto: se mide la satisfacción que le produce el relato al lector.

Para establecer la forma en la que se implementará el modelo en Mexica, es necesario revisar cómo está conformado.

2. Mexica

El sistema escribe historias que son desarrolladas en dos etapas: la primera corresponde a la construcción de la base del conocimiento y la segunda está encargada de construir las historias.

En la primera etapa, el programa obtiene el conocimiento de dos archivos de texto generados por el usuario:

- Diccionario de acciones: se encuentra aquí el conjunto de acciones (odiar, atacar, curar, entre otras) que puede llevar a cabo un personaje. Se asocia con ellas un conjunto de precondiciones, que deben cumplirse para ser utilizadas; por ejemplo, para emplear la acción ‘curar’, el personaje debe estar herido; además, tiene un conjunto de consecuencias o post condiciones que modifican la trama de la historia.
- Las historias previas: representan narrativas hechas por personas, validadas por el autor del sistema.

De ambos archivos se obtiene el conocimiento que será representado a través de estructuras que podrán ser utilizadas como referencia para generar los nuevos relatos. El conjunto de estructuras de conocimiento es llamado conocimiento previo.

En la segunda etapa, con base en el conocimiento previo, el sistema escribe las historias. Para ello, Mexica utiliza dos procesos: *Engagement* y *Reflection*. Durante el estado de *Engagement*, el sistema genera una secuencia de acciones (por ejemplo, la princesa está enamorada del caballero jaguar, el caballero jaguar ama a la virgen, la princesa odia a la virgen). El estado de *Reflection*, evalúa el material generado hasta el momento, y si es necesario lo modifica; además, trata de resolver los *impasses* que surgen durante el desarrollo de la narrativa. Durante la creación de la historia los personajes interactúan, estableciendo ligas emocionales y tensiones, mismas que se describen a continuación.

2.1. Ligas emocionales

Las ligas emocionales se utilizan como precondiciones y post condiciones de las acciones. El sistema hace uso de dos tipos de ellas, principalmente caracterizadas por el tipo de amor que se expresa: el tipo uno representa el amor de hermanos y el tipo dos el amor de pareja. La representación gráfica de las ligas emocionales es a través de flechas. Para el tipo uno la flecha es continua y para el tipo dos la flecha es punteada. Para facilitar el uso de las ligas emocionales dentro del programa, se ha generado una escala donde las emociones irán de -3 a +3. Hay que considerar que los números negativos expresan la emoción negativa (‘odio’), desde poco intenso (-1) hasta muy intenso (-3); mientras que los números positivos expresan a la emoción positiva (‘amor’), desde poco intenso (+1) hasta muy intenso (+3).

A continuación se presentan algunos ejemplos gráficos de ligas emocionales que surgen al momento de ejecutarse una acción.

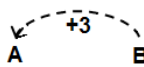
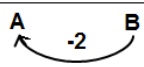
Princesa	A	
Caballero águila	B	
Acción	Liga emocional	Descripción
La princesa está enamorada del caballero águila		Emoción de tipo 2 con intensidad +3
El caballero águila está celoso de la princesa		Emoción de tipo 1 con intensidad -2

Fig. 1. Representación gráfica de algunas ligas emocionales.

2.2. Tensiones

Otro elemento de gran relevancia en la estructura de las historias son las tensiones; por ejemplo, cuando la vida o salud de algún personaje está en riesgo, cuando dos personajes están enamorados de un tercer personaje, cuando muere un personaje, entre otras. Al igual que las ligas emocionales, las tensiones surgen como consecuencia de ejecutar alguna acción; sin embargo, no todas las acciones las disparan. La representación gráfica de las tensiones se hace a través de flechas acerradas, algunas tensiones se presentan en la Figura 2.

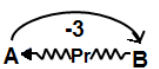

Príncipe	A	
Enemigo	B	
Tlatoani	C	
Acción	Representación	Descripción
El enemigo rapta al príncipe		Emoción de tipo 1 con intensidad -3 y tensión de prisionero
El Tlatoani hiere al enemigo		Emoción de tipo 1 con intensidad -3 y tensión de salud en riesgo

Fig. 2. Representación gráfica de algunas tensiones.

Hasta aquí se han revisado en forma general los elementos que conforman el sistema Mexica, ahora es necesario presentar cómo está conformado el razonamiento analógico, para después plantear el modelo que hace uso de la analogía al momento de surgir un *impasse*.

3. Razonamiento analógico

Siguiendo a Gentner (1983), Lawson (1993), Newton y Newton (1995), se define que la analogía es: la comparación entre dos dominios: uno más conocido y otro poco conocido, que comparten información de tipo relacional.

Los autores Hall (1989), Kedar-Cabelli (1988), Holyoak et al. (1989), proponen que el razonamiento analógico puede ser dividido en tres procesos:

- **Recuperación** (o selección) de una representación del análogo más conocido, de entre un número de posibilidades disponibles en una base de conocimientos, dada una representación del análogo poco conocido.
- **Mapo** entre los elementos que representan el análogo más conocido y los elementos que representan el análogo poco conocido.
- **Transferencia** desde el análogo más conocido al análogo poco conocido, de la estructura y el significado del contenido.

Expuesto el panorama general de lo que es el RA, se presenta cómo está conformado el modelo.

4. Modelo computacional de RA

El modelo comprende dos etapas: identificación de dominios y uso de analogía. La base del conocimiento que contiene las historias creadas por personas representa el dominio más conocido. En cambio, la narrativa que está en proceso de escritura representa el dominio menos conocido. Ambos están establecidos en el ámbito de las narrativas.

Una vez establecidos los dominios, es necesario identificar los elementos que relacionará la analogía. Para ello, se consideran los siguientes puntos:

- Las historias están formadas por una secuencia de acciones.
- Cada acción puede tener asociada un conjunto de emociones o tensiones, ninguna o ambas.
- Tomar en cuenta el momento en el que surge el *impasse* al generar la historia.

Con base en los puntos anteriores, los elementos a relacionar del dominio poco conocido, estarán formados por las emociones y tensiones que se disparan como resultado de ejecutar la acción en el momento que surge el *impasse*. El dominio más conocido se obtiene de una forma similar al del dominio poco conocido, sólo que se construye a partir de las emociones y tensiones que surgen de las acciones que se encuentran en las historias previas.

Si para resolver el problema sólo se considera la última acción en la que surge el *impasse*, y ésta se busca en el dominio más conocido (conjunto de historias previas), únicamente se obtendría una analogía entre acciones; la intención es ir más allá, es decir, saber qué produjo la última acción, obtener más información. Por ejemplo, la muerte de un personaje puede originarse por lesiones de un accidente, por otro personaje que lo atacó o porque fue sacrificado. Por esta razón, se considera adecuado tomar las últimas dos acciones consecutivas, generando pequeños casos concretos que ayudan a tener una visión más amplia de lo que está ocurriendo en la narrativa y con ello obtener una analogía más adecuada.

Los casos contendrán la secuencia de dos acciones representadas en términos de emociones y tensiones. Para ello, se genera una nueva estructura, llamada Elemento de Transición (ET). Esta estructura contiene las ligas emocionales y tensiones que surgen al pasar de una acción a otra de manera consecutiva. Por ejemplo, si una historia previa contiene esta secuencia de acciones: la virgen curó al caballero jaguar y la virgen se enamoró del caballero jaguar, en la primera acción se dispara una emoción de amor de hermanos intenso, que representa el agradecimiento del caballero jaguar hacia la virgen por haberlo curado, y estas ligas emocionales son las que se guardan en el ET; en la segunda acción surge una emoción de amor de pareja intenso; al igual que la acción anterior, las emociones se guardan en otro ET, que es consecutivo al anterior. Por otra parte, también es creada una estructura que agrupa los ET, con la finalidad de tener identificada la forma en que cambian las emociones y tensiones a través del tiempo en toda la historia previa. Esta estructura se denomina Secuencia de Transformación (ST). La Figura 3 presenta un ejemplo de emociones y tensiones que surgen como resultado de la interacción entre los personajes; es decir, cada personaje adquiere el conocimiento de lo que ocurre a su alrededor en términos de emociones y tensiones al ejecutarse cada acción, Esto es con la finalidad de enseñar cómo se obtienen los elementos que conforman cada ET y asimismo la construcción del ST.

virgen	A		
caballero jaguar	B		
Acción	emociones y tensiones		Estructuras de conocimiento ST
	virgen	caballero jaguar	
1 La virgen CURÓ al caballero jaguar			
2 La virgen SE ENAMORÓ del caballero jaguar			

Fig. 3. Representación de las estructuras de conocimiento.

Teniendo establecidos los dominios y sus elementos, únicamente falta utilizar la analogía; por ello, se establecen los tres procesos del razonamiento analógico:

1. En la recuperación el objetivo es seleccionar del dominio más conocido dos estructuras consecutivas que sean análogas en cuanto a emociones y tensiones, con las últimas dos estructuras que se generaron al momento de surgir el *impasse*, las cuales representan el dominio poco conocido.
2. El mapeo consiste en relacionar las emociones y tensiones que existen en las estructuras ST que conforman ambos dominios, más conocido y poco conocido; por consiguiente, los elementos de ambos dominios se agrupan de tres formas:
 - (a) Elementos de relación presentes en ambos dominios.
 - (b) Los elementos del dominio poco conocido que no están presentes en el dominio más conocido.
 - (c) Los elementos del dominio conocido que no están presentes en el dominio poco conocido.

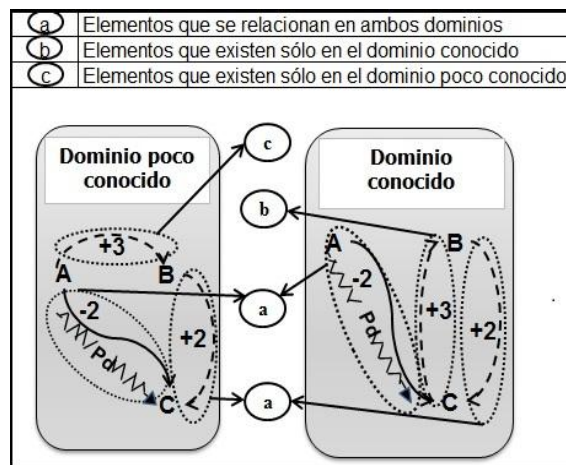


Fig. 4. Mapeo entre dominios.

El mejor caso es cuando todos los elementos quedan en el primer grupo, haciendo una analogía perfecta, pero no siempre es así, habrá casos en donde los elementos estén en las últimas dos agrupaciones, es decir sin relación; por lo que, se evalúa qué tan fuerte es la analogía para tomar una decisión en caso de tener varias alternativas. Por esta razón, se establece una tabla con los posibles elementos que pueden quedar sin relación asignándoles un valor en porcentaje que será restado al cien por ciento de una analogía perfecta ver Tabla 1.

Tabla 1. Valores para los elementos no relacionados.

Elemento sin relación	Porcentaje
Tensiones disparadas por postcondición	20
Tensiones que dispara Mexica	15
Emociones	10
Emociones derivadas por personajes vinculados	5

- La transición se da una vez que se elige la analogía y se procede a determinar cuál es la siguiente acción con la que se debe continuar la historia. Para esto, se establece una relación entre los ET y las estructuras de conocimiento propias de Mexica, ya que en estas estructuras se encuentran las posibles siguientes acciones.

5. Implementación del modelo

En módulo de *Reflection* es donde el modelo de RA se implementa, teniendo como resultado dos alternativas para enfrentar el problema del *impasse*. A continuación se presenta un fragmento de historia donde surge el *impasse*, y es resuelto a través del RA. La creación de la narrativa inicia con la acción, ‘la virgen curó al caballero Jaguar’, en ese momento el sistema dispara una emoción intensa de amor de hermanos, representando el agradecimiento por haberlo curado. En la segunda acción, Mexica decide utilizar la acción, ‘el caballero jaguar asalta a la virgen’; ésta dispara un odio intenso de la virgen hacia el caballero jaguar junto con una tensión de posible peligro; también surge un odio del caballero jaguar hacia sí mismo, por asaltar a quien lo curó. En este momento al no encontrar en su conocimiento alguna acción que le permita continuar la narrativa, surge el *impasse* (ver Figura 5).

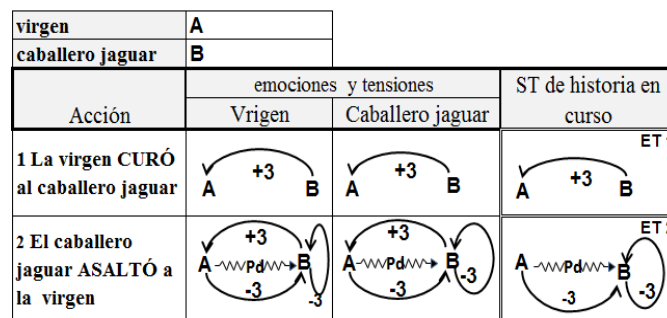


Fig. 5. Surgimiento de *impasse*.

El modelo RA busca la estructura análoga que pase de un momento de amor fraternal a un momento de odio. En este caso se encuentran dos posibles estructuras análogas en el dominio más conocido:

- La primera solución se encuentra en la Secuencia de Transformación dos, específicamente en los ET 5 y ET 7. Las estructuras cambian de un amor fraternal a un odio (relación 1 y 2). La analogía se muestra en la Figura 6.

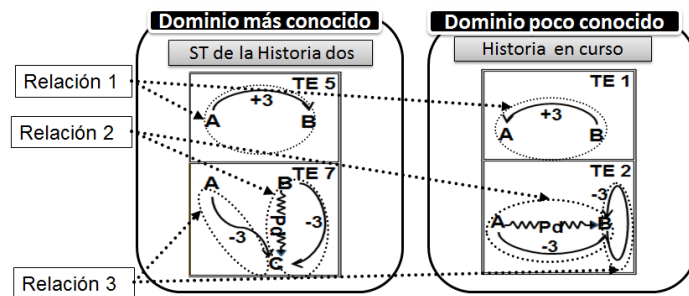


Fig. 6. Primera situación análoga.

- La segunda solución encontrada está en la Secuencia de Transformación tres, en los ET 3 y ET 5. Esta estructura de igual forma es análoga. Es importante mencionar que en esta analogía hay dos personajes ligados con el personaje afectado; como ambos cumplen la misma función, se toma como si fuera uno solo (relación 3). Además, en el dominio origen sobra la tensión Pr; por consiguiente, en la evaluación tiene menor grado de similitud; dicha solución se muestra en la Figura 7.

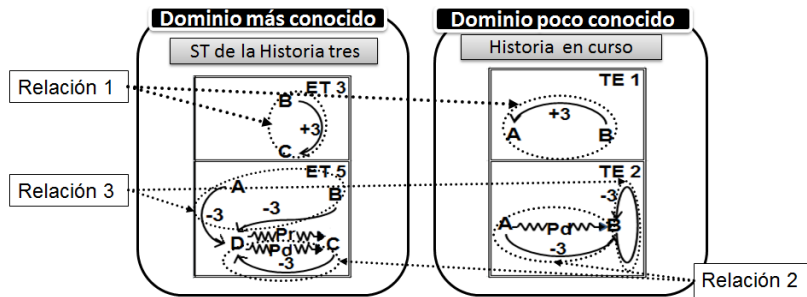


Fig. 7. Segunda situación análoga.

De acuerdo con la agrupación de elementos que conforman los dominios y la tabla de elementos sin relación presentada anteriormente, la primera analogía tiene todos sus elementos relacionados; por lo tanto, tiene un grado de similitud de 100%, mientras que la segunda tiene una tensión de prisionero sin relacionar; por consiguiente, al 100% se le resta el 20% correspondiente al valor de la tensión sin relación, dando como resultado un grado de similitud de 80 %; por lo tanto, la primera analogía es más adecuada para solucionar el *impasse*. Ahora sólo falta determinar cuál es la siguiente acción. Se busca, para ello, el vínculo que hay entre el ET y las estructuras

propias de Mexica. Es en estas últimas donde se encuentran las siguientes posibles acciones; en este caso, corresponde a una estructura con una tensión y tres ligas emocionales, que propone como siguiente acción ‘atacar’, por lo que se continua la narrativa escribiendo: ‘la virgen atacó al caballero jaguar’.

6. Evaluación del modelo

La evaluación del modelo de razonamiento analógico es realizada en dos fases: la primera consiste en verificar que el modelo de RA cumpla con el propósito de minimizar el número de *impasses* que provocan historias inconclusas. Para cumplir con dicho objetivo, se generaron en el sistema Mexica diez casos con *impasse*, y para resolverlos se utilizaron ambas alternativas: método propio y razonamiento analógico. Por un lado, el método propio no tuvo éxito dado que no pudo resolver ninguno de los *impasses*, por otro lado al hacer uso de RA se pudieron resolver seis de los diez casos; con ello se pudo comprobar que el RA es una herramienta apta para la solución del *impasse*.

La segunda fase consistió en verificar que la acción que resuelve el *impasse* propuesta por el RA, fuera tan coherente e interesante como la que se genera a través del método propio de Mexica.

Para evaluar esta fase se utilizó una muestra conformada por 30 sujetos, 13 mujeres y 17 hombres de entre 19 y 25 años ($M = 21$), que fueron seleccionados de forma no probabilística, en una universidad pública al norte del Distrito Federal. Se utilizó un instrumento que fue desarrollado para el presente estudio, que constó de seis reactivos tipo viñetas con formato de respuesta tipo *Likert* de cinco puntos, en los que se contemplaron seis elementos primordiales de la narrativa: fluidez, estructura, contenido, suspenso, calidad y gusto, con el objetivo de determinar, a través de ellos, la coherencia y el interés de las historias generadas por el RA. Además, es considerado el momento (al principio, a la mitad y al final) en que aparece el *impasse* al generar la narrativa.

A través de los resultados se observó que los sujetos tenían diferentes opiniones de las alternativas según el momento en el que aparece el *impasse*. Por un lado, les gustan más las historias resueltas con el modelo de razonamiento cuando surge el *impasse* al principio y al final (ver Figura 8); sin embargo, cuando surge a la mitad de la narrativa el relato gusta más utilizando el MP (ver Figura 9). Por otro lado, señalan que las historias generadas utilizando ambas alternativas mantienen la coherencia e interés. No obstante al observar los indicadores en conjunto existen ciertas diferencias. Por ello, se realizó un análisis estadístico a través de la prueba *t* de Student, resultando en que no hay diferencias estadísticamente significativas al iniciar ($t(29) = .914$, $p > 0.05$), a la mitad de la historia ($t(29) = .868$, $p > 0.05$), ni al final ($t(29) = .029$, $p > 0.05$); dicho de otra forma, ambas herramientas son igual de efectivas para resolver el problema del *impasse*; sin embargo, si el análisis se hace por cada variable, la estructura es la única variable que tiene una diferencia significativa cuando el *impasse* surge al final de la narrativa (ver Figura 8).

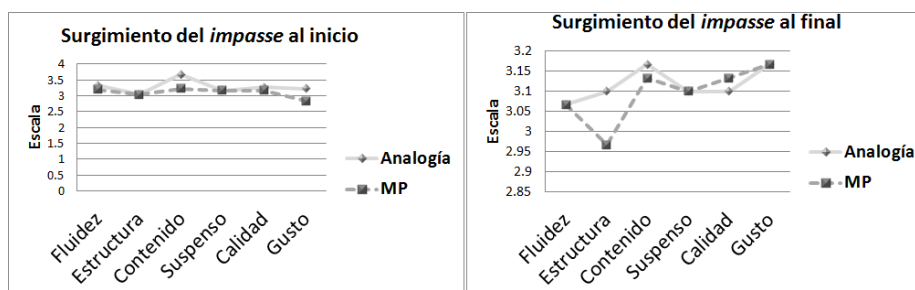


Fig. 8. Representación gráfica del surgimiento del *impasse* al inicio y al final.

En la gráfica de la Figura 9 se observó que los sujetos de estudio aprecian que hay diferencia en la variable de estructura. Utilizando la analogía mantiene un buen grado de estructura; sin embargo, disminuye el grado haciendo uso del método propio de Mexica. Probablemente esto se deba a que son pocas las estructuras de conocimiento con las que cuenta el sistema para finalizar la historia.

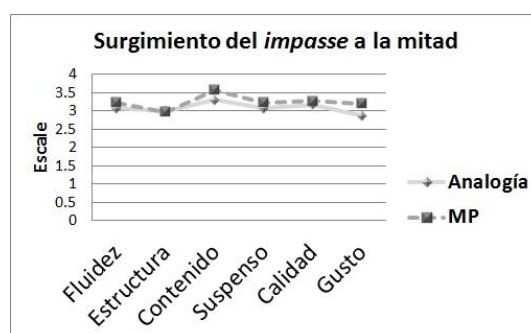


Fig. 9. Representación gráfica del surgimiento del *impasse* a la mitad.

7. Conclusiones

El objetivo de la presente investigación ha sido crear un modelo computacional basado en el razonamiento analógico, que pudiera aplicarse en los sistemas generadores de historias como una alternativa que apoyara el desarrollo del relato, con lo cual se pudo concluir que sí es factible crear este tipo de modelos. El modelo de razonamiento analógico logra resolver seis de cada diez casos de *impasse* que el método propio de Mexica no resuelve, resultando benéfico ya que reduce a 40% el número de historias inconclusas. Para reducir más el porcentaje de historias inconclusas es necesario hacer más re combinaciones de estructuras de conocimiento; en el presente proyecto sólo se crearon casos con dos estructuras consecutivas, pudiéndose tomar tres, o más estructuras consecutivas para generar la analogía.

Las variables: fluidez, estructura, contenido, suspenseo, calidad y gusto, permitieron hacer una adecuada evaluación de las historias en cuanto a coherencia e interés. Los resultados de la evaluación sugieren que Mexica puede utilizar el RA, cuando el *im-*

passé surge al inicio o al final de la narrativa, dado que en todos los casos la evaluación realizada al modelo de RA fue igual o superior a la del método propio de Mexica. A través de los resultados estadísticos se concluye que el modelo de razonamiento analógico es adecuado como una alternativa que apoya en la generación de narrativas. Mexica ahora es uno de los primeros en explorar la creación de narrativas utilizando un modelo de razonamiento analógico, mostrando ser apto en los casos que logra resolver. Por último, el modelo ayuda al entendimiento de la analogía a través de procesos cognoscitivos.

Referencias

1. Dartnall, T.: Creativity, Cognition and Knowledge: An Interaction. London: Praeger (2002)
2. Gentner, D.: Structure-Mapping: a Theoretical Framework for Analogy. Cognitive Science. The MIT Press. Vol. 7, pp. 155-170 (1983)
3. Hall, R.: Computational approaches to analogical reasoning: A comparative analysis. Artificial Intelligence, 39-120 (1989)
4. Hofstater, D., & Mitchell, M.: The copycat project: A model mental fluidity and analogy-making. (D.a. In Hofstadter, Ed.) Basic Books. (1995)
5. Hofstater, D.: Fluid Concepts and Creative Analogies: Computer Models of the Fundamental Mechanisms of Thought. Basic Books (1995)
6. Holyoak, K. & Hummel, J.: Understanding analogy within a biological symbol system. In D. Gentner, K.J. Holyoak and B.N. Kokinov (Eds.), The Analogical Mind: Perspectives from Cognitive Science. London: The MIT Press, 161-195 (2002)
7. Holyoak, K. J. & Thagard P.: Analogical mapping by constrain satisfaction. Cognitive Science. 13 (1989)
8. Kedar-Cabelli, S.: Analogy - From a unified perspective. Analogical Reasoning, ed. D.A. Helman, Kluwer Academic Publishers. Dordrecht, 65-103 (1988)
9. Koestler, A.: The Act of Creation. London: Hutchinson. (1964)
10. Lawson, A. E.: The Importance of Analogy: A Prelude to the Special Issue. Journal of Research in Science Teaching. 30 (10), 1213-1214 (1993)
11. Mitchell, M.: Analogy Making as Perception: A Computer Model. Cambridge, Mass.: Bradford Books. (1993)
12. Newton, D. & Newton, L. Using Analogy to Help Young Children Understand. Educational Studies, Vol. 21 (3), 379-393 (1995)
13. Pérez, R.: MEXICA: a Computer Model of Creativity in Writing. PhD dissertation. University of Sussex (1999)
14. Pérez, R.: Employing Emotions to Drive Plot Generation in a Computer-Based Storyteller. Cognitive Systems Research. Vol. 8, number 2, 89-109 (2007)
15. Perkins, D.: The Mind's Best Work. Cambridge, Mass.: Harvard University Press. (1981).
16. Salvucci, D. D., & Anderson, J. R.: Integrating analogical mapping and general problem solving: The path-mapping theory. Cognitive Science 25, 67-110 (2001)

Extracción automática de hechos en libros de texto basada en estructuras sintácticas

Honorato Aguilar-Galicia¹, Grigori Sidorov¹ y Yulia Ledeneva²

¹ Centro de Investigación en Computación, Instituto Politécnico Nacional, México
aguilargh@hotmail.com, sidorov@cic.ipn.mx

² Universidad Autónoma del Estado de México, México
yledeneva@yahoo.com

Resumen Las oraciones se conforman de fragmentos de texto que pueden separarse y poseer independencia semántica, pero normalmente se encuentran fusionados en la oración, enunciando algo de una forma más amplia. Las oraciones de forma conjunta le dan sentido a un tema, o lo que llamamos información. A estos fragmentos de texto que obtenemos de relacionar entidades a través de un solo verbo, en nuestra investigación les llamamos “hechos”. Por ejemplo en la oración “*Los mesopotámicos nos legaron la rueda y la escritura*”, podemos identificar los siguientes: “*Los mesopotámicos legaron la rueda*” y “*Los mesopotámicos legaron la escritura*”. En estos hechos las entidades son “*mesopotámicos*”, “*rueda*” y “*escritura*”; y el verbo es “*legaron*”. En el presente artículo exponemos lo qué es un hecho, la importancia de extraerlos, y proponemos un método para extraerlos de manera automática con base al análisis de estructuras sintácticas.

Palabras clave: Recuperación de información, extracción de información, extracción de hechos, estructuras sintácticas.

1. Introducción

Desde tiempos antiguos la humanidad ha generado información la cual ha guardado en forma de lenguaje natural, ya fuera en arcilla o papiros. En la actualidad se continúa generando información la cual se guarda en libros, revistas, periódicos.

Con el invento de las computadoras y su desarrollo intenso, hemos logrado que la información actualmente se guarde en forma de bits, es decir, la información se guarda en formato electrónico. Y con el invento de la red Internet y la Web, la cantidad de información crece casi de forma exponencial, lo cual en ocasiones dificulta su administración, pues con tanta información es difícil encontrar la que se busca.

Áreas como la “Recuperación de información” y la “Extracción de información” nos ayudan a encontrar la información que necesitamos. Dentro de esta última podemos ubicar otra área que tratamos en este artículo: “La extracción de información semántica” o lo que llamamos aquí, de forma resumida, “La extracción de hechos”.

Un texto está conformado por párrafos y cada párrafo es un conjunto de oraciones. Estas últimas se conforman de pequeñas unidades, que de forma conjunta le dan sentido a la oración, pero que podrían separarse y poseer independencia semántica, pero

normalmente se encuentran fusionadas en la oración, enunciando algo de una forma más amplia.

A estas pequeñas unidades las llamamos “hechos”. La Real Academia Española define *Hecho* como: “*m. Acción u obra, m. Cosa que sucede, m. Asunto o materia de que se trata*” [RAE22] y [Joo07] nos dice que existen dos tipos de hechos: los simples y los complejos. Por ejemplo dos hechos simples pueden ser: Pedro y 1989, también llamados entidades. Y un hecho complejo es la relación que se establece entre estas entidades por medio de un verbo, entonces tenemos: *Pedro nació en 1989*. En [Her11] encontramos que “*se considera como hecho a una porción de texto, generalmente más pequeña que una oración, la cual tiene independencia semántica y que contiene un solo verbo asociado a entidades*”.

Veamos un ejemplo, en la oración “*Benito Juárez nació en San Pablo Guelatao, Oaxaca, en 1806*”, podemos identificar dos hechos:

- *Benito Juárez nació en San Pablo Guelatao, Oaxaca.*
- *Benito Juárez nació en 1806.*

Observamos claramente que son fragmentos de la oración; cada uno nos enuncia algo de forma independiente, o sea, ninguno de ellos necesita al otro para transmitir su información; en cada hecho notamos a un solo verbo, un sujeto y un complemento que juntos forman a una pequeña unidad con sentido completo.

2. La utilidad de los hechos

Contar con una base de datos de hechos es muy importante para otras áreas del procesamiento de lenguaje natural [Mon01], ya que estos pueden ser consultados para realizar sus tareas. Áreas como sistemas de pregunta-respuesta y creación de resúmenes automáticos aprovechan los hechos ya almacenados.

- Sistemas de Pregunta-Respuesta

IBM ha estado construyendo Watson, un proyecto llamado DeepQA [Watson], es un sistema de preguntas y respuestas. Watson la computadora se enfrenta con gran éxito a humanos en el famoso juego de TV en EEUU llamado Jeopardy. Para la adquisición de conocimientos Watson visita bases de datos, taxonomías, ontologías; y los datos recolectados los almacena en sus bases de datos nombradas “fuentes de respuestas y evidencias” [Dav10]. Así que nuestro propósito de extraer hechos es para almacenarlos en una base de datos que posteriormente puedan ser empleados por un sistema como Watson.

- Resúmenes Automáticos

Una base de datos de hechos puede ocuparse para crear resúmenes automáticos abstractivos principalmente. Por ejemplo, si en la base de datos tenemos los siguientes hechos “Gato es una mascota”, “Canario es una mascota”, “Pedro compró un gato”, y “María compró un canario”. Los algoritmos del método que está creando el resumen abstractivo, pueden aprovechar estos hechos para inferir, generalizar o parafrasear conocimiento; obteniendo como resultado: “Pedro y María compra-

ron una mascota cada uno”. La extracción de hechos de un texto también crea un resumen extractivo de ese texto, pero es un resumen muy sencillo ya que en él puede haber redundancia de información.

3. Estado del arte

3.1. Extracción de hechos con intervención de usuario y entrenamiento

En [Joo07] la extracción de hechos lo hace marcando dentro de un texto lo que llama entidades o hechos simples (ejemplo: Williams y 1980) y luego en una ontología se establece una relación entre estos hechos, a lo que llaman hecho complejo o “hecho” (Williams was born on 1980). Para esto ocupan un software de anotación y una ontología.

3.2. Extracción y fusión de hechos de documentos múltiples

En este trabajo se extrae hechos realizando un cruce y fusión de información de diferentes documentos [Mon08], aprovechando la redundancia de datos. Se trabaja con software de anotación y ontologías [Man06].

Los dos trabajos anteriores trabajan con textos en inglés. Nuestra investigación es con textos en español, además el enfoque es con base al análisis de estructuras sintácticas, y los hechos se guardan en una base de datos relacional. No ocuparemos software de anotación, ni una ontología para establecer relaciones que identifiquen los hechos.

3.3. Un sistema de extracción automática de información semántica de los libros de textos estructurados

En [Her11] el trabajo de tesis extrae hechos basándose en análisis de estructuras sintácticas. El analizador sintáctico que se ocupa es “Connexor”, un software que tiene un costo para adquirirlo, además de que es un software que tiene un desempeño poco regular en análisis de oraciones en español.

Aquí se extrajeron hechos de 50 oraciones, ya que Connexor en ocasiones arrojaba árboles mal formados. Los cuales se utilizaron para aplicar heurísticas en la búsqueda de hechos. Los hechos extraídos se almacenaron en una base de datos relacional.

En nuestra investigación utilizamos FreeLing-3.0 como analizador sintáctico, un software que no tiene ningún costo por usarlo [FrLi30]. Además fue creado especialmente para analizar las lenguas de Europa, incluyendo el idioma Español.

3.4 Un esquema de evaluación semiautomática

En muchas tareas de procesamiento de lenguaje natural, nos encontramos con el problema de determinar el nivel de granularidad adecuado para las unidades de información. Por lo general, usamos frases para modelar piezas individuales de información. Sin embargo, más aplicaciones de Procesamiento de Lenguaje Natural (PLN)

nos obligan a definir las unidades de texto más pequeñas que frases, esencialmente descomponiendo sentencias dentro de una colección de frases.

Cada frase lleva una pieza independiente de información que puede ser utilizado como una unidad independiente. Estas unidades de información de mayor granularidad son usualmente llamadas como “nuggets” [Lia07].

Trabajos anteriores muestran que las personas pueden crear nuggets de una manera relativamente sencilla. Sin embargo, la creación automática de los nuggets no es trivial.

Con base a un análisis manual y modelado computacional de nuggets, se definen como sigue:

- Un nugget es un evento o una entidad.
- Cada nugget consiste en dos partes: el ancla y el contenido.
El ancla es:
 - El sustantivo principal de la entidad.
 - El verbo principal del evento, más el sustantivo núcleo de su entidad asociada (si más de una entidad está unida al verbo, entonces su sujeto).

El contenido es una pieza única y coherente de la información asociada con el ancla. Cada anclaje debe tener varios contenidos distintos.

Cuando un nugget contiene sentencias anidadas, esta definición es aplicada recursivamente.

4. Método propuesto

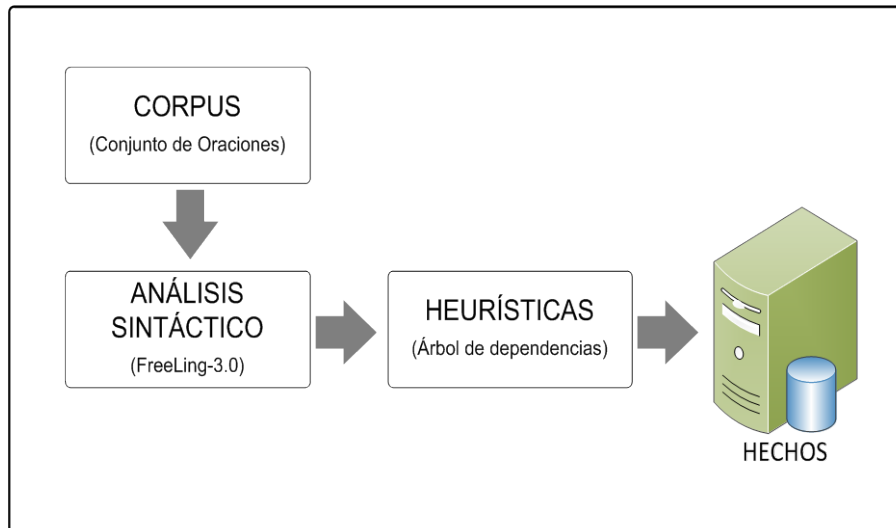


Fig. 1. Método propuesto.

La Figura 1 muestra las partes que conforman al método propuesto y como se relacionan.

4.1. Corpus

Se utilizó un corpus compuesto por 55 oraciones en español, este conjunto de oraciones fueron tomadas de libros de texto [Her11] y [SEP-H6-10], pues al ser creados para fines educativos contienen gran cantidad de conceptos, y por lo tanto contienen una gran cantidad de hechos. Además, estos libros tratan sobre temas muy específicos.

Los libros de texto, por su naturaleza, están estructurados por temas y capítulos. En donde cada capítulo está compuesto por un título, un conjunto de párrafos y cada párrafo por un conjunto de oraciones. A esta composición le llamamos un texto estructurado. Y fue esta característica la razón principal de seleccionar los libros de texto, pues las oraciones de un capítulo pueden separarse y ser procesadas por el método propuesto como una unidad.

Estas 55 oraciones fueron seleccionadas porque contiene una secuencia gramatical aceptable para el método, esto es, contienen sujeto y predicado. Oraciones formuladas como interjección o pregunta no fueron elegidas.

4.2 Análisis sintáctico

Estructuras sintácticas.

La extracción automática de información semántica que realizamos en nuestra investigación, se basa en el análisis de estructuras sintácticas, específicamente en el análisis de lo que se conoce en gramática como oración, el objeto de análisis en nuestro método.

Una “estructura”, en el diccionario de la lengua española de la Real Academia Española [RAE22] encontramos que “estructura” es “1. f. Distribución y orden de las partes importantes de un edificio”, y “2. f. Distribución de las partes del cuerpo o de otra cosa”. De acuerdo a estas definiciones podemos deducir que una “estructura” para nuestro objeto de análisis sería: “la distribución y orden de las palabras en una oración” y considerando la sintaxis, agregamos que “las palabras son funciones unas de otras”, como dice [Mun00] “las palabras adquieren un significado preciso y cumplen una función sintáctica determinada”:

- Se lastimó la muñeca izquierda mientras jugaba a la pelota.
- La muñeca que le regalé a mi hija cierra los ojos.

Aisladamente, la palabra *muñeca* tiene varias acepciones, pero en cada oración toma una de ellas; además, esta misma palabra cumple una función distinta, en la primera oración es objeto directo y en la segunda, es sujeto [Rio09].

Dentro de las distintas partes de la gramática, la sintaxis es la que se dedica al estudio de la oración. Su estudio se basa en las diferentes funciones que desempeñan los componentes de la oración. [Fue10]

[Mun00] nos dice que “la sintaxis es la parte de la gramática que estudia la manera como se combinan y ordenan las palabras para formar oraciones; analiza las funciones que aquellas desempeñan, así como los fenómenos de concordancia que pueden presentarse entre sí”.

Herramienta FreeLing.

Para cada una de las oraciones realizamos un análisis sintáctico de forma automática, para ello utilizamos FreeLing-3.0, configurado con la opción “Dependency Parsing”, la cual realiza un análisis morfosintáctico y nos proporciona un árbol de dependencias.

Cada nodo del árbol de dependencias representa una palabra de la oración, que contiene información sintáctica y morfológica de cada una de ellas, organizada de forma jerárquica.

Para el etiquetado morfosintáctico, FreeLing-3.0 utiliza un conjunto de etiquetas propuesto por el grupo [EAGLES]. Sirve para la anotación morfosintáctica de lexicones y corpus para todas las lenguas europeas, y por lo tanto para el español [Eagv20].

Características de FreeLing-3.0.

Es una suite de código abierto que sirve para realizar análisis morfosintáctico de texto, publicado bajo la licencia GNU General Public License de la Free software Foundation [FrLi30].

Algunas características:

- Tokenización de texto.
- Análisis morfológico.
- Reconocimiento de fechas, números, monedas y medidas físicas (velocidad, peso, temperatura, densidad, etc.)
- PoS tagging.
- Análisis de dependencia basado en reglas.
- Resolución de referencia nominal.

4.3. Heurísticas

El método propuesto utiliza heurísticas para identificar los hechos. Estas trabajan sobre los árboles de dependencia que nos proporciona [FrLi30], en donde cada nodo cuenta con las siguientes etiquetas de la palabra que contiene. La descripción de las etiquetas es parte de la documentación de [FrLi30].

- Func: Indica la función que tiene la palabra, por ejemplo sujeto (subj), dobj (objeto directo).
- Synt: Indica la relación sintáctica, por ejemplo sintagma nominal (sn), grupo preposicional (grup-sp).
- Form: La forma real de la palabra, tal como está escrita en la oración.
- Lemma: Lemma de la palabra.
- Tag: Etiqueta de acuerdo a las etiquetas EAGLES [Eagv20].

A continuación describimos las heurísticas empleadas.

Heurística 1: Es la heurística principal, se aplica en el árbol en busca de la siguiente secuencia gramatical: Sujeto – Verbo – Complemento.

1. Ubicar verbo principal en el árbol o subárbol.

2. Buscar y extraer al sujeto. Ubicaremos al sujeto en los nodos dependientes del verbo obtenido en el paso 1, con la siguiente etiqueta: {func = subj} y {synt = sn}. O {func = subj-pac} y {synt = sn}
3. Buscar y extraer al complemento. Estará compuesto por los nodos hermanos siguientes del sujeto, es decir, se toma los nodos posteriores al nodo sujeto.
4. Construir hecho. El hecho estará formado por: (Sujeto extraído en el paso 2) + (Verbo extraído en el paso 1) + (Complemento extraído en el paso 3).

Heurística 2: Aplicada cuando la estructura contiene conjunción copulativa o disyuntiva de sustantivos.

1. Aplicar la heurística 1, para obtener el sujeto y el verbo.
2. Ubicar los sustantivos coordinados. Con las etiquetas {func = co-n} y {synt = sn},
3. Construir hechos. Se construyen varios hechos: (Sujeto obtenidos en el paso 1) + (Verbo obtenido en el paso 1) + (Complemento 1, 2, n obtenido en el paso 2).

Heurística 3: Aplicada cuando la estructura contiene conjunción copulativa o disyuntiva de verbos.

1. Ubicar los verbos coordinados. Con las etiquetas {func = co-v} y {synt = grup-verb}.
2. Aplicar heurística 1 para obtener sujeto y complemento.
3. Construir hechos. Se construye un hecho por cada verbo encontrado en el paso 1: (Sujeto obtenido en paso 2) + (Verbo 1, 2, n obtenido en paso 1) + (Complemento obtenido en paso 2).

Heurística 4: Aplicada cuando la estructura contiene un pronombre relativo.

1. Ubicar pronombre relativo. Con las etiquetas {func = subord-mod}, {synt = subord-rel} y {tag = PRxxxxxx}.
2. El pronombre relativo depende de un nodo sintagma nominal etiquetado con {synt = sn} y {tag = Nxxxxxx}, este será el sujeto para el hecho.
3. El pronombre relativo tiene como descendiente a un verbo. Este será el verbo para el hecho.
4. El complemento lo conformarán los elementos dependientes del verbo del paso 3.
5. Construir hecho con (Sujeto obtenido en paso 2) + (Verbo obtenido en paso 3) + (Complemento obtenido en paso 4).

Heurística 5: Aplicada cuando en la estructura encontramos las preposiciones *desde* y *hasta*.

1. Aplicar la heurística 1 para extraer el sujeto y verbo.
2. Ubicamos la preposición *desde* y la guardamos junto con sus nodos dependientes. Será el complemento de un hecho. El nodo tiene las siguientes etiquetas: {func = cc}, {synt = grup-sp} y {tag = SPS00}.
3. Construimos un hecho: (Sujeto obtenido en paso 1) + (Verbo obtenido en paso 1) + (Complemento obtenido en paso 2).
4. Ubicamos la preposición *hasta* y la guardamos junto con sus nodos dependientes. Será el complemento de otro hecho. El nodo tiene las siguientes etiquetas: {func = ador}, {synt = grup-sp} y {tag = SPS00}.

5. Construimos un hecho: (Sujeto obtenido en paso 1) + (Verbo obtenido en paso 1) + (Complemento obtenido en paso 4).

Heurística 6: Aplicada cuando en la estructura encontramos la preposición *en*.

1. Aplicar la heurística 1 para extraer el sujeto y verbo.
2. Ubicamos la preposición *en* y la guardamos junto con sus nodos dependientes. Será el complemento de un hecho. El nodo tiene las siguientes etiquetas: {func = sp-obj}, {synt = grup-sp} y {tag = SPS00}.
3. Construimos un hecho: (Sujeto obtenido en paso 1) + (Verbo obtenido en paso 1) + (Complemento obtenido en paso 2).
4. Ubicamos la preposición *en* y la guardamos junto con sus nodos dependientes. Será el complemento de otro hecho. El nodo tiene las siguientes etiquetas: {func = sp-obj}, {synt = grup-sp} y {tag = SPS00}.
5. Construimos un hecho: (Sujeto obtenido en paso 1) + (Verbo obtenido en paso 1) + (Complemento obtenido en paso 4).

4.4 Base de datos de hechos

Los hechos obtenidos son almacenados en una base de datos relacional. Esta cuenta con tres tablas: Tabla de entidades, Tabla de verbos y Tabla de relaciones. En la Tabla de entidades guardamos las entidades (Sustantivos, Adjetivos, Preposiciones) de la oración, en la Tabla de verbos guardamos los verbos, y en la tercera tabla guardamos las relaciones entre las entidades y verbos, es decir, lo que forma un hecho.

5. Resultados

Para probar el método propuesto primero se proporcionó a una persona una guía para identificar hechos en una oración y se le encargó que identificara y extrajera los hechos de cada una de las oraciones de nuestro corpus de análisis. Esta persona extrajo un total de 119 hechos. Posteriormente se aplicó el método propuesto a cada una de las oraciones y comparamos ambos resultados.

5.1 Guía para un humano para extraer hechos

A continuación describimos el algoritmo, para que una persona pueda identificar y extraer hechos desde una oración. La estructura de un hecho es: Sujeto – Verbo – Complemento. El algoritmo nos ayuda a identificar los tres elementos y nos indica la estructura del hecho.

1. Localizar el sujeto en la oración. Para hacerlo contestamos a una de las siguientes preguntas: ¿De qué o de quién se habla? o ¿Quién o qué realiza la acción?
2. Localizar el predicado en la oración. Para hacerlo contestamos a una de las siguientes preguntas: ¿Qué se dice, de quien se habla o de lo que se habla? o ¿Qué se dice, de quien o lo que realiza la acción

3. Buscar hechos. La estructura del hecho es Sujeto - Verbo – Complemento, así que se construye el hecho de acuerdo a ella. Identifiquemos las siguientes características en la oración, de forma específica en el predicado:
- (a) PRIMER Hecho. Extraemos el verbo principal, normalmente se encuentra al inicio del predicado. Con el sujeto obtenido en el paso 1, este verbo y el resto del predicado como complemento se crea el primer hecho. Revisamos los siguientes incisos para verificar si la oración en proceso contiene estas características.
 - (b) Revisar perífrasis. Generalmente la perífrasi tiene la forma: (verbo auxiliar) + (preposición o conjunción) + (infinitivo, gerundio o participio). Cuando la oración contenga esta característica, se tomará el último elemento como el verbo para el hecho.
 - (c) CONJUNCION.Copulativa. Conjunciones copulativas son las “que coordinan dos o más palabras las cuales desempeñan una misma función. También pueden unir oraciones. Las conjunciones copulativas son *y, e, ni*” [Mun00]. Ejemplo: “El domingo compré discos de música hindú, turca y rusa”. Si la oración contiene conjunción copulativa, obtendremos un hecho por cada término coordinado por la conjunción. Para el ejemplo anterior los hechos son: "El domingo compré discos de música hindú", "El domingo compré discos de música turca", "El domingo compré discos de música rusa".
 - (d) CONJUNCION.Disyuntiva. Las conjunciones disyuntivas “son conjunciones que enlazan palabras u oraciones para expresar posibilidades alternativas, distintas o contradictorias. Las conjunciones disyuntivas son *o, u*” [Mun00]. Ejemplo: “Pedro se hospedará en una pensión u hotel cualquiera”. Cuando en la oración se tenga conjunción disyuntiva, obtendremos un hecho por cada término coordinado por la conjunción. Para el ejemplo anterior los hechos son: "Pedro se hospedará en una pensión", "Pedro se hospedará en un hotel cualquiera".
 - (e) PRONOMBRE.Relativo. “Los pronombres relativos hacen referencia a alguien o a algo que se ha mencionado antes en el discurso o que ya es conocido por los interlocutores. Los pronombres relativos, funcionan, en la mayor parte de los casos, como elementos de subordinación de oraciones. Los pronombres relativos son *que, quien, quienes, cual, cuales, cuanto, cuantos, cuanta, cuantas.*” [Mun00]. Ejemplo: "Pedro conoció a un estudiante que sabe hablar chino". Como el pronombre relativo hace referencia a alguien o a algo que se ha mencionado antes, entonces buscaremos al sujeto (sustantivo, pronombre personal) para un siguiente hecho, en la parte inmediata que antecede al pronombre relativo. El verbo de este hecho se encuentra localizado después del pronombre relativo. Los hechos del ejemplo, son: "Pedro conoció a un estudiante", "Estudiante sabe hablar Chino".
 - (f) PREPOSICIÓN: *desde y hasta*. “*Desde*. Denota inicio de una acción en el tiempo o en el espacio. *Hasta*. Expresa el fin de algo o límite de lugar, de número o de tiempo” [Mun00]. Ejemplo: “La primavera comprende desde el mes de marzo hasta el mes de junio”. Si una oración contiene estas preposiciones, se

formará un hecho por cada una de ellas. Para el ejemplo tenemos los siguientes hechos: "La primavera comprende desde el mes de marzo", "La primavera comprende hasta el mes de junio".

- (g) PREPOSICIÓN: *en*. La preposición *en* "indica tiempo, expresa lugar, señala modo, significa ocupación o actividad, indica medio o instrumento, forma locuciones adverbiales" [Mun00]. Ejemplo: "Pedro estudia *en* la mañana y *en* la noche". Cuando una oración que contenga una o más de una preposición *en*, se formará un hecho por cada una de ellas. Los hechos del ejemplo son: "Pedro estudia *en* la mañana", "Pedro estudia *en* la noche".

4. Fin.

5.2 Evaluación del método propuesto

Abajo presentamos unos ejemplos de oraciones analizadas.

Algunas oraciones que fueron analizadas con nuestro método. Las dos primeras fueron analizadas con éxito, pues cumplen una cierta estructura sintáctica en la cual se basa el método. En la tercera y cuarta nuestro método presenta dificultades.

Oración 1: "*La numeración arábiga procede de India*". De aquí se obtuvo el hecho: "*Numeración arábiga procede de India*", se tuvo éxito porque se identifica claramente el sujeto, el predicado incluyendo al verbo.

Oración 2: "*La civilización China nos heredó el papel, la pólvora, una forma de imprenta rudimentaria, y la brújula*". Sus hechos: "*Civilización china heredó el papel*", "*Civilización china heredó pólvora*", "*Civilización china heredó forma de imprenta rudimentaria*", "*Civilización china heredó brújula*". Aquí se obtiene el mismo sujeto y verbo para todos los hechos, y el complemento de los hechos son términos coordinados por una conjunción.

Ahora veamos las oraciones en donde nuestro método tiene complicaciones.

Oración 3: "*Roma no imponía ideas políticas o credos en sus territorios*". El método propuesto no incluye en los hechos la palabra "*no*", entonces al no contemplarse nos da como resultados hechos con un significado contrario a lo que expresa la oración: "*Roma imponía ideas políticas*", "*Roma imponía credos*".

Oración 4: "*El mamut era un animal de gran tamaño al que se cazaba mediante diversas técnicas*". Los hechos obtenidos por la persona son: "*El mamut era un animal*", "*El mamut era de gran tamaño*", "*El mamut se cazaba mediante diversas técnicas*". El método tiene dificultad para manejar la preposición "*de*" pues no la toma como otro hecho, los hechos que obtiene son: "*El mamut era un animal de gran tamaño*", "*El mamut se cazaba mediante diversas técnicas*", o sea dos.

Usamos las siguientes medidas de evaluación.

En la evaluación del método con el corpus, este ha obtenido un total de 116 hechos, de los cuales 82 fueron correctos y 34 incorrectos. A continuación las medidas de evaluación.

$$\text{Precisión del método} = \frac{\text{Hechos correctos obtenidos por el método}}{\text{Total de hechos obtenidos por el método}} \quad (1)$$

$$\text{Recall} = \frac{\text{Hechos correctos obtenidos por el método}}{\text{Total de hechos existentes en el texto (extraídos por un humano)}} \quad (2)$$

$$\text{F1} = (2 * \text{Precisión del método} * \text{Recall}) / (\text{Precisión del método} + \text{Recall}) \quad (3)$$

Los resultados que hemos obtenido hasta el momento son los siguientes:

Tabla 1. Resultados de la evaluación.

Característica	Resultado
Precisión del método	70.6%
Recall	68.9%
F1	69.7%

6. Conclusiones

Es importante decir que nuestra investigación está en proceso, así que los resultados hasta el momento no son definitivos pero son buenos para continuar en esta línea. Hemos identificado ciertos patrones en la estructura sintáctica de las oraciones cuando nos enuncian algo. Una unidad semántica sólo expresa información cuando nos dice algo de alguien. Estos patrones los hemos plasmado en heurísticas en el método, ayudándonos a obtener información semántica desde las oraciones y por ende de los textos.

Es importante mencionar que nuestra investigación también comprende otras áreas de estudio, como es la generación automática de resúmenes, pues nos encontramos que al extraer las unidades mínimas de información en una oración, realmente estamos extrayendo las unidades que informan algo en un texto. Si unimos todas estas unidades informativas o lo que nosotros llamamos hechos, obtenemos un resumen del texto de donde se hayan obtenido las oraciones.

Finalmente, también es importante mencionar que los resultados obtenidos hasta el momento son con base al análisis de nuestro corpus actual, corpus que acrecentaremos para identificar más patrones, y de esta forma mejorar nuestros resultados y nuestras heurísticas.

Agradecimientos. Trabajo realizado con el apoyo parcial del gobierno de México (proyectos CONACYT 50206-H y 83270, SNI) e Instituto Politécnico Nacional, México (proyectos SIP 20111146, 20113295, 20120418, COFAA, PIFI), Gobierno del DF (ICYT-DF proyecto PICCO10-120) y la Comisión Europea (proyecto 269180).

Referencias

- [Dav10] David Ferrucci, Eric Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya A. Kalyanpur, Adam Lally, J. William Murdock, Eric Nyberg, John Prager, Nico Schlaefer, and Chris Welty. Building Watson: An Overview of the DeepQA Project. Association for the Advancement of Artificial Intelligence. ISSN 0738-4602 (2010)
- [EAGLES] Expert Advisory Group on Language Engineering Standards <<http://www.ilc.cnr.it/EAGLES96/home.html>> [Consulta: 30/04/2012]
- [Eagv20] Introducción a las etiquetas eagles (v. 2.0). <<http://nlp.lsi.upc.edu/freeling/doc/tagsets/tagset-es.html>> [Consulta: 30/04/2012]
- [FrLi30] FreeLing 2.2. An Open Source Suite of Language Analyzers. <<http://nlp.lsi.upc.edu/freeling/index.php>> [Consulta: 30/04/2012]
- [Fue10] Fuentes de la Corte, Juan Luis. Gramática Moderna de la lengua española. Limusa, México (2010)
- [Gel09] Gelbukh, A. & Sidorov, G. Procesamiento automático del español, con enfoque en recursos léxicos grandes. Instituto Politécnico Nacional. México (2009)
- [Her11] Herrera de la Cruz, Juve Andrea. Sistema de extracción automática de información semántica de los libros de textos estructurados. Tesis de maestría, CIC-IPN, México (2011)
- [Hov06] Hovy, Eduard. Learning by Reading: An Experiment in Text Analysis. Springer-Verlag Berlin Heidelberg (2006)
- [Joo07] Jooose, W. User Trainable Fact Extraction. Masters Thesis, Universiteit Twente de ondernemende universiteit (2007)
- [Lia07] Liang Zhou, Namhee Kwon, and Eduard Hovy. A Semi-Automatic Evaluation Scheme: Automated Nuggetization for Manual Annotation. Proceedings of NAACL HLT 2007, Companion Volume, pages 217–220, Rochester, NY. Association for Computational Linguistics (2007)
- [Man06] Mann, Gideon S. Multi-Document Statistical Fact Extraction and Fusion. Doctor Thesis, The Johns Hopkins University, Baltimore, Maryland (2006)
- [Mor04] De la Mora Alejandro. Las partes de la oración. Trillas. México (2004)
- [Mun00] Munguía Zatarain Irma, Munguía Zatarain Martha Elena, Rocha Romero Gilda. Gramática Lengua Española. Reglas y ejercicios. Ediciones Larousse, S.A. de C.V. México, D. F. (2000)
- [SEP-H6-10] Secretaría de Educación Pública. Historia Sexto grado. ISBN: 978-607-469-414-7. México, D.F. (2010)
- [Watson] Watson Research Center. IBM, <www.watson.ibm.com/index.shtml> [Consulta: 30/04/2012]
- [Mon01] Manuel Montes y Gómez, Alexander Gelbukh, Aurelio López López. Mining the news: trends, associations, and deviations. Computación y Sistemas, Vol. 5 N 1, pp. 14-24 (2001)
- [Rio09] Miguel Ángel Ríos Gaona, Alexander Gelbukh, Sivaji Bandyopadhyay: Web-based Variant of the Lesk Approach to Word Sense Disambiguation. In: MICAI 2009. Proc. of 2009 Eighth Mexican International Conference on Artificial Intelligence, IEEE CS Press, pp. 103–107 (2009)
- [Mon08] Alfredo Monroy, Hiram Calvo, Alexander Gelbukh: Using Graphs for Shallow Question Answering on Legal Documents. In: MICAI 2008. Lecture Notes in Artificial Intelligence N 5317, Springer, 165-173 (2008)

Agrupación de documentos utilizando representaciones holográficas reducidas

Norma L. Cuautle-Rivera¹, Maya Carrillo¹ y Aurelio López-López²

¹ Facultad de Ciencias de la Computación, BUAP,
Av. San Claudio y 14 Sur Ciudad Universitaria, 72570 Puebla, México
n_lucero_c@hotmail.com, cmaya@cs.buap.mx

² Coordinación de Ciencias Computacionales, INAOE,
Luis Enrique Erro 1, Sta.Ma. Tonantzintla, 72840, Puebla, México
alopez@inaoep.mx

Resumen En este artículo proponemos la utilización de la representación holográfica reducida (HRR) para agrupar documentos de texto. Las HRRs son una representación novedosa de documentos que capturan información sintáctica de los mismos, la cual es producida utilizando la metodología del espacio vectorial conocida como Indización Aleatoria. Se empleó el conjunto de datos Reuters y se compararon los resultados con trabajos reportados en la bibliografía. Los resultados obtenidos muestran que las HRRs mejoran la tarea de agrupamiento con respecto a la representación vectorial empleando el mismo algoritmo de agrupamiento y tienen un desempeño competitivo con respecto a otros métodos de agrupamiento reportados, todo esto con una eficiencia en tiempo y espacio notablemente menor.

Palabras clave: Agrupamiento de documentos, modelo vectorial, indización aleatoria, representaciones holográficas reducidas.

1. Introducción

El ser humano emplea el lenguaje para expresar ideas y comunicarse con otros. La comprensión del lenguaje es compleja, debido a la variación y ambigüedad inherentes a él. Estas características dificultan el procesamiento automático del lenguaje, que se complica aún más porque la computadora no tiene aún la capacidad de comprensión del lenguaje que tenemos los humanos. En dicha comprensión interviene el contexto y la abstracción de ideas para desambiguar e interpretar de la mejor manera el significado de diferentes planteamientos. En las siguientes oraciones se ilustra la ambigüedad en distintos niveles del lenguaje:

- a. A nivel léxico: Tomó una botella y se fue (¿bebió la botella o la tomó con la mano?).
- b. A nivel morfológico: Nosotros plantamos papas (¿estamos en el proceso de plantar o ya se plantaron?).
- c. A nivel sintáctico: Veo al gato con el telescopio (¿uso telescopio para ver al gato o veo al gato que tiene el telescopio?).

- d. A nivel semántico: Todos los estudiantes de la escuela hablan dos lenguas (¿cada uno habla dos lenguas o sólo se hablan dos lenguas determinadas?).

El procesamiento del lenguaje natural (NLP) se apoya principalmente del análisis morfológico, léxico y sintáctico. Tradicionalmente, los documentos se representan como una lista de términos léxicos independientes, que son representados como vectores. Dichos vectores combinados linealmente permiten representar documentos. Esta representación se conoce como modelo de espacio vectorial (VSM), con el inconveniente de que en ella, se pierden las relaciones existentes entre palabras y, por lo tanto, la identificación de conceptos importantes que podrían ayudar a representar de manera más adecuada los documentos. Por otra parte cuando se utiliza el VSM los documentos y términos se representan en una matriz de $\mathbf{n} \times \mathbf{m}$, donde \mathbf{n} es el número de términos y \mathbf{m} el número de documentos. Dado que no todos los términos aparecen en todos los documentos, esta matriz generalmente es muy dispersa y de dimensión grande, por lo que para procesarla es necesario emplear técnicas de reducción de dimensión. Uno de los métodos empleados es la descomposición en valores singulares (SVD) que es cara en términos de tiempo de procesamiento y memoria requerida. En el presente trabajo se experimenta con una técnica de reducción de espacio vectorial conocida como Indización Aleatoria (RI), que ha demostrado ser útil en tareas de recuperación de información y clasificación [5] [6] [7]. Con la ayuda de RI se construyen Representaciones Holográficas Reducidas (HRRs) [1] [2] [3] con el propósito de incluir información sintáctica en la representación de los documentos para determinar el efecto que dicha representación tiene en la tarea de agrupamiento de documentos. El resto del documento está organizado de la siguiente manera: en la sección 2 se explica la metodología de indización aleatoria; en la sección 3 se describe la representación holográfica reducida para documentos, en la 4 se mencionan algunos trabajos relacionados, en la sección 5 se describen los experimentos realizados; en la 6 se presentan los resultados obtenidos y finalmente la sección 7 presenta las conclusiones y posible trabajo futuro.

2. Indización aleatoria

La indización aleatoria (RI), es una metodología cuya idea básica es acumular vectores de contexto, basándose en la ocurrencia de las palabras en contextos. Esta técnica es incremental y no requiere una fase de reducción de dimensión [4]. La técnica de RI se describe en los siguientes pasos:

1. A cada contexto (documento) se le asigna una representación única y generada aleatoriamente, la cual es llamada vector índice. Estos vectores índice son dispersos, de alta dimensión, y ternarios, lo que significa que su dimensión (d) es del orden de miles, y que consisten de un pequeño número de +1s y -1s aleatoriamente distribuidos, con el resto de los elementos de los vectores iguales a 0.
2. Posteriormente se construyen vectores de contexto para cada palabra, mediante el recorrido del texto, cada vez que una palabra ocurre en un documento, el vector índice de dicho documento, se agrega al vector de contexto

para la palabra en cuestión. Las palabras son representadas por vectores de contexto d -dimensionales que son la suma de los contextos en los que aparecen dichas palabras.

3. Una vez contruidos los vectores de contexto para el vocabulario (palabras diferentes en una colección de documentos), los documentos se representan como la suma ponderada de los vectores de contexto de las palabras que aparecen en ellos.

3. Representaciones holográficas reducidas

Las HRRs son vectores n -dimensionales, cuyos elementos siguen una distribución normal $N(0,1/n)$. Dichos vectores permiten codificar estructura textual utilizando representaciones distribuidas. Para tal propósito se emplea el operador de convolución circular para relacionar términos, el cual es apropiado por las siguientes razones: al operar dos vectores se obtiene un vector del mismo tamaño, lo que permite su utilización en procesos recursivos sin incrementar la dimensión del espacio vectorial, además preserva la similitud estructural [1].

La convolución circular (\otimes) mapea dos vectores n -dimensionales a un vector \mathbf{z} . Si \mathbf{x} y \mathbf{y} son vectores n -dimensionales, entonces los elementos de $\mathbf{z} = \mathbf{x} \otimes \mathbf{y}$ son definidos como:

$$z_i = \sum_{k=0}^{n-1} x_k y_{i-k} \quad i = 0 \text{ to } n - 1 \text{ (donde los subíndices son módulo-}n\text{)} \quad (1)$$

Para representar los documentos empleando HRRs se siguieron los siguientes pasos:

- a. Se generaron vectores de contexto para el vocabulario de la colección utilizando la indización aleatoria.
- b. Los documentos se etiquetaron por categorías sintácticas.
- c. Se creó la representación HRR para cada término utilizando la convolución circular para relacionar el término con su categoría sintáctica. Por ejemplo, el par **deal/VBP** se codificó creandole un HRR \mathbf{z} donde el vector \mathbf{x} es el vector de contexto de **deal** y \mathbf{y} un HRR generado aleatoriamente. Para cada categoría (etiqueta) sintáctica se generó un HRR aleatorio.
- d. Los vectores (HRRs) resultantes se ponderaron empleando el esquema *tf-idf* (que considera la importancia de los términos dentro de los documentos y de la colección) y se sumaron para representar documentos.

Si bien la HRR se ha utilizado para recuperar analogías [2], en clasificación [7] y recuperación de información [5], hasta donde se tiene conocimiento, no se ha utilizado para realizar agrupamiento de documentos.

4. Trabajo relacionado

Guan R. et al en [9] presentan un nuevo algoritmo de agrupamiento de texto llamado propagación de afinidad mediante semillas (en inglés Seeds Affinity

Propagation, SAP). Mencionan dos principales contribuciones, la primera es una nueva métrica de similitud, la cual captura información estructural de los textos; y la segunda es un método de construcción de semillas para mejorar el proceso de agrupamiento semi-supervisado. Utilizan el corpus Reuters y realizan una comparación de los resultados con los algoritmos k-Means y AP (Affinity Propagation). Obtienen mejores resultados con respecto a los algoritmos con que se comparan utilizando F-measure y la entropía como métricas de comparación. La información estructural que consideran mejora los resultados de agrupamiento empleando la medida de similitud que proponen, llamada Tri-conjunto (Tri-Set) en lugar del Coseno. Otro factor que contribuye a mejorar los resultados son las semillas seleccionadas, las cuales se obtienen de las características representativas de los objetos etiquetados.

Zimmerling M. en [10] emplea dos algoritmos llamados k-Means++ y KKz como extensiones del algoritmo k-Means básico, debido a que estos mejoran la elección de los centroides iniciales, mediante técnicas de semillas aleatorias. Para realizar las evaluaciones de los algoritmos utilizaron los conjuntos de datos Classic3, Reuters-21578 y F-measure como métrica de evaluación. Los algoritmos propuestos no mejoraron substancialmente al algoritmo k-Means en cuanto a tiempo de ejecución o número de iteraciones, debido a la naturaleza de los datos.

Cleuziou G., en [11] presenta un nuevo enfoque para determinar posibles cubiertas de datos para la agrupación de grupos traslapados. Propone un nuevo algoritmo, de nombre OKM, el cual es una generalización de k-Means. OKM consiste de una nueva función objetivo para minimizar bajo restricciones de multi-asignación, es decir, se explora el espacio de cubiertas en vez del espacio de particiones a diferencia de k-Means. Menciona que la tarea de asignar cada dato a uno o varios grupos no es una tarea trivial, por lo cual propone la heurística de desplazarse a través de la lista de grupos prototipo desde el más cercano al más lejano, y asignar cada vector mientras su imagen es mejorada, la nueva asignación es conservada si es mejor que la anterior, asegurándose de que disminuya la función objetivo. Los experimentos se realizaron con el corpus Reuters y se evaluaron con la métrica F-measure. Sus resultados muestran un comportamiento consistente del algoritmo OKM al proveer mejores grupos traslapados.

Su Z. et al, en [12], muestran un nuevo algoritmo de agrupamiento híbrido, el cual se basa en el algoritmo de cuantización vectorial (vector quantization, VQ) y el de estructura de crecimiento de celdas (growing -cell structure, GCS). Emplean VQ para mejorar la salida de agrupación de GCS y mejorar el problema de entrenamiento insuficiente. Los pesos de los nodos de salida del agrupamiento GCS son considerados como los vectores prototipo iniciales de VQ, después de varias ejecuciones de entrenamiento, los nuevos pesos de los vectores prototipo ganadores de VQ reemplazarán a los pesos de los nodos de GCS. Tal proceso puede considerarse como una fase de entrenamiento adicional en la agrupación de GCS y resuelve el problema de entrenamiento insuficiente. Se denota como D el conjunto de documentos y F al conjunto de vectores de características de cada documento en D . Para cada nuevo documento insertado, no se requiere entrenar

la red desde el principio, debido a que las actualizaciones se pueden realizar basándose sobre resultados previos. Todos los documentos del corpus Reuters se ponderaron utilizando $tf-idf$, y se emplearon los algoritmos K-Means, VG, GCS y la propuesta híbrida. De acuerdo los resultados de los experimentos, el método propuesto alcanza mejor desempeño que los métodos con los que se comparan.

Ayad, et al. en [13], proponen combinar agrupaciones producidas por diferentes técnicas de agrupamiento para descubrir los tópicos de los documentos de texto, la agregación de estas agrupaciones genera mejores estructuras de datos. Después de que se forman los grupos de documentos, se emplea un proceso llamado extracción de tópicos, el cual selecciona los términos del espacio de características (es decir, el vocabulario de la colección entera) para describir el tópico de cada grupo, en esta etapa se re-calculan los pesos de los términos de acuerdo a la estructura de los grupos obtenidos. Para representar a los documentos se usó el modelo del espacio vectorial, para el pesado de los términos, se empleó $tf-idf$. Para la agrupación por agregación, se emplearon algoritmos jerárquicos, incrementales y particionales. Se utilizó F-measure para evaluar y comparar los tópicos extraídos y la calidad de la agrupación antes y después de la agregación. La evaluación experimental mostró que la agregación puede mejorar exitosamente tanto la calidad de la agrupación como la precisión de los tópicos comparándose con las técnicas de agrupación individuales.

5. Experimentos

Para evaluar el efecto de las HRRs en la tarea de agrupamiento, se utilizó el conjunto de documentos Reuters-21578 [8], el cual es popular en la comunidad de procesamiento de lenguaje natural, pues está constituido por noticias de diversos contextos que presentan diferentes características.

El corpus Reuters es una colección de 21,578 artículos que apareció en el servicio de noticias Reuters en 1987, distribuido sobre 116 categorías. Cada documento fue manualmente clasificado por los editores del servicio de noticias. El número de documentos asignado a cada categoría varía, algunas categorías tienen un gran número de documentos, como la llamada *earn*, mientras que otras categorías, como *rye*, tienen muy pocos documentos.

El primer experimento se realizó tomando el total de documentos de la colección con la misma cantidad de clases en los conjuntos de entrenamiento y prueba de Reuters, 68 clases. Posteriormente, con el fin de compararnos con los resultados reportados en [11], se utilizó un subconjunto de documentos distribuidos en 10 clases para contar con un total de 3696 documentos únicos, tratando de aproximarnos lo más posible al subconjunto de documentos utilizados en [11], hasta donde los detalles de dicha fuente nos permitió hacerlo, Tabla 1.

Dado que el corpus Reuters tiene clases muy desbalanceadas, se seleccionaron grupos de documentos para formar cinco conjuntos de documentos como se describe en [9]. Para lo cual se seleccionaron 800 documentos de texto, contenidos en 10 clases, para cada caso. La distribución de los diferentes números de documentos entre las 10 clases se muestra en la Tabla 2.

En Reuters los documentos de las clases *corn*, *grain* y *wheat* son muy similares entre sí, lo cual hace que sean muy difíciles diferenciarlos. La distribución de estas clases puede influir profundamente en los resultados del agrupamiento. Para cada uno de los cinco casos, las clases *corn*, *grain* y *wheat* contienen los siguientes porcentajes de documentos 10 %, 30 %, 50 %, 70 % y 90 %, respectivamente.

Se realizó el pre-procesado de cada documento del corpus etiquetando los términos con MontyLingua [14] para obtener las categorías sintácticas de los términos, se eliminaron los símbolos de puntuación y palabras vacías, y se truncaron las palabras utilizando el algoritmo de PorterStemmer.

Una vez preprocesados los documentos, se utilizó la indización aleatoria para generar las HRRs, empleando vectores de dimensión 1024.

Los experimentos fueron ejecutados en una PC Intel(R) Core(TM)2 Quad CPU 2.86 GHz con 4 GB de RAM. Se empleó la herramienta Weka 3.6.4 para realizar el agrupamiento de documentos y el algoritmo k-Means.

Tabla 1. Subconjuntos de documentos considerado en los primeros experimentos.

<i>Conjunto de datos</i>	<i>Documentos</i>	<i>Clases</i>
Total de documentos	9592	68
Subconjunto	3696	10

Tabla 2. Porcentajes de documentos por cada clase para los 5 casos considerados.

<i>Casos</i>	% <i>acq</i>	% <i>corn</i>	% <i>crude</i>	% <i>earn</i>	% <i>grain</i>	% <i>interest</i>	% <i>money-fx</i>	% <i>ship</i>	% <i>trade</i>	% <i>wheat</i>
Caso 1	30	2.5	20	15	5	10	7.5	5	2.5	2.5
Caso 2	10	10	10	10	10	10	10	10	10	10
Caso 3	8.8	16.3	7.5	6.3	16.3	5	3.8	2.5	16.3	17.5
Caso 4	7.5	23.1	6.3	5	23.1	3.8	1.3	2.5	3.8	23.8
Caso 5	2.5	30	1.3	1.3	30	1.3	1.3	1.3	1.3	30

6. Resultados

En la Tabla 3 se presentan los resultados obtenidos al comparar la propuesta de representación para los documentos, con los resultados obtenidos por diferentes métodos reportados en la bibliografía, hasta donde fue posible aproximar el número de clases y documentos. Puede verse que aparentemente los HRRs no mejoran los resultados del agrupamiento, sin embargo las grandes diferencias

que existen en las clases de la colección puede ser la principal causa. Tratando de homogeneizar el conjunto de pruebas se construyeron subconjuntos de documentos de acuerdo al trabajo de Guan R. et al [9]. En la Tabla 4 se presenta una comparación de los resultados obtenidos, al emplear las HRRs para representar documento y el algoritmo k-Means para clasificarlos, con los obtenidos en dicho trabajo. Puede observarse que la combinación de HRRs con k-Means se desempeñó mejor que el algoritmo AP cuando se utiliza el coseno como métrica de similitud, e incluso que el algoritmo AP con la métrica de similitud Tri-set propuesta en [9]. También mejoró al algoritmo SAP cuando se utiliza el coseno como métrica de similitud. Por otro lado, dicho algoritmo combinado con la métrica Tri-set mejora los resultados obtenidos con las HRRs. Mejora, que con mucha probabilidad se debe a la diferencia en las métricas de similitud empleadas, distancia euclidiana en k-Means.

El agrupamiento empleando HRRs, para representar los documentos y k-Means para agruparlos, mejora en cuanto a F-measure a k-Means cuando los documentos se representan con el VSM en un 48.63 % en promedio para los cinco casos considerados; en un 20. % a los obtenidos con AP y coseno; en un 27.70 % a los obtenidos con AP y Tri-Set y en un 7.72 % a SAP con coseno. Sin embargo es superado por SAP(Tri-set) en un 8.11 %.

Tabla 3. Resultados obtenidos por diferentes propuestas de agrupamiento

<i>Referencia</i>	<i>HRR</i>					
	<i>Doc.</i>	<i>Clases</i>	<i>F-measure</i>	<i>Doc.</i>	<i>Clases</i>	<i>F-measure</i>
1.An extended version of the k-Means method for overlapping clustering [11]	1308	10	0.76	3696	10	0.36
2.An Empirical Study of K-Means Initialization Methods for Document Clustering [10]	8193	65	0.35	9592	68	0.31
3.Topic discovery from text using aggregation of different clustering methods [13]	3000	10	0.39	3696	10	0.36

7. Conclusiones y trabajo futuro

Hemos presentado una representación novedosa para capturar la estructura sintáctica de documentos empleando la indización aleatoria. Los resultados mostraron que la representación es capaz de agrupar documentos de manera más precisa que cuando estos se representan con el VSM. Así mismo mostraron su competitividad con respecto a otros trabajos reportados en la bibliografía.

Tabla 4. Comparación de resultados con el trabajo relacionado, utilizando F-measure

<i>Referencia</i>	<i>Algoritmo</i>	<i>Caso 1</i>	<i>Caso 2</i>	<i>Caso 3</i>	<i>Caso 4</i>	<i>Caso 5</i>	<i>Prom.</i>
Text	K-MEANS	0.518	0.397	0.368	0.280	0.269	0.366
Clustering	AP(CC)	0.450	0.450	0.450	0.450	0.450	0.450
with Seeds	AP(Tri-Set)	0.577	0.482	0.419	0.364	0.290	0.426
Affinity	SAP (CC)	0.662	0.519	0.511	0.450	0.385	0.505
Propagation [9]	SAP(Tri-Set)	0.749	0.606	0.573	0.544	0.489	0.592
HRR-1024	K-MEANS	0.571	0.567	0.624	0.542	0.414	0.544

Utilizar RI y HRRs para representar documentos permite reducir la matriz de términos contra documentos en estos experimentos a únicamente una matriz de 800 por 1024 lo que optimiza el tiempo de cualquier algoritmo de agrupamiento. Para estos experimentos, el tiempo promedio fue de 0.26 min.

El trabajo en proceso nos permitirá probar la representación con otros algoritmos de agrupamiento y con diferentes métricas a fin de determinar de manera precisa el beneficio de la representación propuesta para el agrupamiento de documentos. También estamos por realizar un análisis cualitativo de la utilidad de la información sintáctica en la discriminación entre grupos, dado que aporta mayor información que el indexado tradicional en SVM.

Agradecimientos. Esta investigación fue financiada por el proyecto PROMEP /103.5/11/4481. El segundo y tercer autor parcialmente financiados por SNI México.

Referencias

1. Plate, T.A.: Holographic Reduced Representation: Distributed representation for cognitive structures. CSLI Publications, (2003)
2. Plate T.A : Analogy Retrieval and Processing with Distributed Vector Representation, Expert Systems, 17: 1, pp. 29-40,(2000)
3. Plate T. A. : Distributed Representation in: Encyclopedia of Cognitive Science, Macmillan Reference Ltd, (2002)
4. Sahlgren, M.: An Introduction to Random Indexing, In: Methods and Applications of Semantic Indexing Workshop at the 7th Int. Conf. on Terminology and Knowledge Engineering (2005)
5. Carrillo M., Eliasmith C., and Lopez-Lopez A., Combining Text Vector Representations for Information Retrieval, In: Text, Speech and Dialogue. Procs. of the 12th International Conference Text, Speech and Dialogue, LNAI, 5729 pp. 24-31,(2009)
6. Carrillo M., Villatoro-Tello E., López López A., Eliasmith C., Villaseñor-Pineda L., Montes-y-Gómez M., Concept based representations for ranking in geographic information retrieval, In: Procs. of the 7th international conference on Advances in natural language processing, IceTal, pp. 85-96. (2010)

7. Sahlgren, M., Cöster R.: Using Bag-of-Concepts to Improve the Performance of Support Vector Machines in Text Categorization. In: Procs. of the 20th Int. Conf. on Computational Linguistics, pp. 487- 493 (2004)
8. Lewis D.D.: Reuters-21578 Text Categorization Test Collection. Available at http://www.daviddlewis.com/resources/testcollections/reuters_21578, May (2004)
9. Guan R., Shi X., Marchese M., Yang C., and Liang Y.: Text Clustering with Seeds Affinity Propagation. In: IEEE Trans. Knowledge and Data Eng., 23:4, pp.627-637, (2011)
10. Zimmerling M.: An Empirical Study of K-Means Initialization Methods for Document Clustering, Dresden University of Technology, Germany) at <http://www.tik.ee.ethz.ch/marcoz/other/clustering08.pdf>, (2008)
11. Cleuziou G.: An extended version of the k-Means method for overlapping clustering. In: 19th ICPR Conference, Tampa, Florida, USA, pp. 1–4, (2008)
12. Su Z., Zhang L., Pan Y.: Document Clustering Based on Vector Quantization and Growing-Cell Structure, Developments. In: 16th international conference on industrial and engineering applications of artificial intelligence and expert systems, IEA/AIE 2003, Loughborough,UK, pp.326-336, (2003)
13. Ayad H., Kamel M.:Topic Discovery from Text Using Aggregation of Different Clustering Methods. In: Procs. of the 15th Conference of the Canadian Society for Computational Studies of Intelligence on Advances in Artificial Intelligence Springer-Verlag London, UK, (2012)
14. Liu, Hugo: MontyLingua: An end-to-end natural language processor with common sense. web.media.mit.edu/~hugo/montylingua (2004)

Evaluación de características para el proceso de atribución de autoría

Esteban Castillo, Darnes Vilariño, David Pinto, Maya Carrillo
e Iván Olmos

Benemérita Universidad Autónoma de Puebla,
Facultad de Ciencias de la Computación,
Av. San Claudio y 14 Sur, Puebla, México
ecjbuap@gmail.com, {darnes, dpinto, cmaya, iolmos}@cs.buap.mx

Resumen La detección de autoría es una tarea de categorización que intenta descubrir el nombre del autor para un cierto documento anónimo dado como entrada. Para resolver esto, se necesita determinar primordialmente el conjunto de características que distinguen el modo de escritura de cada autor. En el presente trabajo se discuten los resultados obtenidos al evaluar diversas características para el proceso de atribución de autoría. El estudio se ha realizado utilizando tres corpora distintos: los dos primeros fueron proporcionados en el foro denominado “Uncovering Plagiarism, Authorship, and Social Software Misuse Lab (PAN 2011)”, mientras que el tercero es un corpus balanceado que proviene de un trabajo altamente citado en el estado del arte, conteniendo documentos de noticias corporativas industriales. De acuerdo a los experimentos realizados, las características elegidas permiten representar documentos escritos por diferentes autores, de tal manera que un clasificador supervisado basado en máquinas de soporte vectorial puede asignar, en promedio, la autoría de un cierto documento anónimo con un 60% de exactitud.

Keywords: Atribución de autoría, extracción de características, clasificación, máquinas de soporte vectorial

1. Introducción

En la forma más básica, el problema de atribución de autoría consiste en determinar de manera unívoca el autor correcto de un documento anónimo. En años recientes se ha incrementado el número de investigaciones en este sentido, y el problema ha sido enmarcado dentro del modelo o paradigma de categorización de textos, es decir, dicha problemática ha sido llamada de distintas maneras tales como adjudicación de textos, determinación de autoría, o como actualmente se le conoce atribución de autoría.

La búsqueda de características para representar a un autor específico supone una barrera que depende del tipo de lenguaje utilizado en cada texto. Si bien el lenguaje está limitado por el tema tratado o por el público a quién va dirigido (entre otros factores), el autor aun es libre de escribir como desea hacerlo, (lo cual

supone un obstáculo más en la detección de dicho autor), es decir, un autor es libre de hacer oraciones o párrafos de la longitud que desee, de usar el vocabulario y los signos de puntuación como lo considere conveniente, o de usar el tiempo verbal que más le agrade. Lo anterior nos lleva al hecho de que a pesar de haber muchos elementos impredecibles en la detección de autoría (como un conjunto de autores abierto), la búsqueda de características que cree o plasmen fielmente los principales rasgos de un autor, es una cuestión primordial en el proceso de entrenamiento sobre el mismo.

La presente investigación tiene como objetivo mostrar los resultados obtenidos al detectar los autores reales de un conjunto de documentos, considerando la representación tanto de los documentos de entrenamiento, como los de prueba utilizando 18 características seleccionadas (léxicas y sintácticas). El trabajo de investigación se desarrolló usando tres corpora conformados de diferente manera. El objetivo fue observar el comportamiento de las características seleccionadas sobre colecciones de textos que no comparten el dominio ni el tipo de escritura.

Es por ello que nos hemos planteado la siguiente pregunta de investigación: En que medida es posible detectar los rasgos distintivos de cada autor, independientemente del corpus con que se trabaje?

El resto de este artículo está estructurado de la siguiente manera: En la sección 2 se discuten los resultados reportados hasta el momento en la literatura para darle solución a este tipo de problema. En la sección 3 se presentan las características a evaluar y el proceso de clasificación. Por último, se comenta la evaluación realizada y se dan las conclusiones obtenidas en esta investigación.

2. Trabajos previos

Las investigaciones mas relevantes realizadas hasta el momento se encaminan fundamentalmente hacia el tipo de características que realmente se deben utilizar, es decir, si deberían ser características léxicas, sintácticas o semánticas. Adicionalmente, ciertos resultados importantes se han desarrollado buscando la mejor forma de representar los documentos, considerando algunas de las características mencionadas con anterioridad. En este sentido, a continuación se presentan los trabajos más recientes con respecto a la extracción y uso de características en la representación de los elementos de escritura distintivos de un cierto autor.

Stamatatos [6], por ejemplo, presenta un estudio acerca de los elementos más importantes a tomar en cuenta en la tarea de atribución de autoría. El estudio es bastante completo y considera una revisión del estado del arte de este problema desde sus inicios hasta la actualidad, haciendo énfasis en la extracción de características y de cómo éstas ayudan en el proceso de detección de un autor. En este trabajo, Stamatatos presenta los siguientes tipos de características como la base de los elementos más usados para representar el estilo: características estilométricas, léxicas, semánticas, de carácter y de estructura. El estudio muestra las dificultades y los retos aún sin resolver en el área de atribución de autoría.

En [2], se hace una revisión de las principales diferencias en los distintos tipos de características presentes en los textos. Se proponen también medidas para obtener el grado de acierto con respecto a las técnicas usadas para atribuir el autor de un documento anónimo dado.

Shlomo [1], por otro lado, presenta un resumen de la evaluación de la metodología aplicada para la atribución de autoría en el congreso PAN 2011 mostrando las distintas soluciones propuestas por los participantes del foro para resolver este problema. Dentro del análisis hecho en ese trabajo se pueden identificar cuatro clases de elementos fundamentales con respecto a las características usadas: Características derivadas del uso de las palabras en el texto (Palabras, N-gramas de palabras, Pronombres, Palabras del discurso, Contracciones y abreviaciones), Características a nivel de carácter (N-gramas de caracteres, Sufijos, Prefijos, Puntuación, Longitud de sentencia y palabras), Características basadas en el formato del texto (Ortografía, Orden de las estructuras gramaticales) y Características basadas en la sintaxis del texto (Partes del discurso, Sintaxis).

En [3] se plantea la idea de usar diferentes clasificadores para atribuir la autoría de un documento anónimo. Cada clasificador se entrena con una sola característica y posteriormente se establece un sistema de “votación”, en donde un conjunto de características final es determinado como representativo de un autor si y solo si dicho conjunto presenta un comportamiento óptimo en los distintos clasificadores. Entre las características más importantes se usan las siguientes: Número de palabras, Número de líneas de texto, Número de caracteres en un documento, Número de sentencias, Número de bloques de texto, Número de palabras cerradas y Uso de las letras capitales. La relevancia de ese artículo radica en la combinación de distintos clasificadores para probar el peso y el grado de utilidad de un conjunto de características con respecto a un autor dado.

Si bien, existen otros trabajos en la literatura relacionados con la atribución de autoría, hasta donde sabemos, los anteriores resultan ser los más representativos.

En general, los resultados que se han obtenido hasta el momento muestran que las características seleccionadas por los autores depende mucho del corpus con el que se está trabajando. Es por ello que en la presente investigación se propone trabajar con 17 características sobre tres corpora que poseen estructuras estilométricas diferentes. El objetivo principal es analizar si las características propuestas logran detectar los rasgos distintivos de cada autor, independientemente de las estructuras textuales existentes en los diferentes corpora.

3. Extracción de características y proceso de clasificación

Una de las principales preocupaciones en la atribución de autoría es la búsqueda de todas aquellas propiedades cuantificables de un autor, que sean capaces de diferenciarlo de otros. A este tipo de elementos presentes en la mayoría de los textos se les llama características. Es de suma importancia el determinar las características adecuadas, y de preferencia óptimas que califiquen el estilo de escritura de cada autor.

3.1. Características escogidas

Para la extracción de las características representativas se consideró conservar intacto el texto de cada autor, en este caso, únicamente se eliminaron las etiquetas XML presentes en los textos. Se usa un corpus de entrenamiento que asocia a cada documento su correspondiente autor, con la finalidad de entrenar un modelo supervisado. La descripción del corpus de entrenamiento y del modelo de clasificación se explica más adelante. A continuación se presentan y describen las características a evaluar en este trabajo de investigación.

- Características a nivel de oración:
 - Palabras con frecuencia igual a 1. Son aquellas palabras que aparecen una sola vez en cada documento que pertenece al mismo autor.
 - Palabras con frecuencia igual a 2. Son aquellas palabras que aparecen dos veces en cada documento que pertenece al mismo autor.
 - Palabras más frecuentes. Las 10 palabras que aparecen con mayor frecuencia en los documentos de un autor dado.
 - Colocaciones. Los pares de palabras que siempre aparecen juntas en un documento dado (bigramas)¹.
 - Prefijos de palabras. La subcadena que antecede a la base léxica.
 - Sufijos de palabras. La subcadena que precede a la base léxica.
 - Trigramas de palabras. Todas las subsecuencias de tres palabras de cada documento.
 - Palabras cerradas. Las palabras con poco contenido semántico, como artículos, pronombres, preposiciones, etc.
 - Número de palabras por oración. Se cuantifica para cada documento el número de palabras que posee cada oración.
 - Número de oraciones. Se cuantifica cuantas oraciones posee cada documento.
 - Trigramas de las categorías gramaticales del documento. Todas las secuencias de 3 categorías que aparecen a lo largo de todo el documento.
- Características a nivel de caracter:
 - Combinación de vocales. Se elimina de cada palabra las consonantes y se considera la combinación de vocales restantes como una característica (se unen todas las vocales repetidas en cada combinación).
 - Permutación de vocales. Se elimina de cada palabra las consonantes y se consideran todas las combinaciones de vocales, cada combinación como una característica.
 - Letras del abecedario. Se considera el número de veces que aparece cada letra del abecedario en un documento dado.
 - Trígrama de caracter. Se cuantifican todas las subsecuencias de tres caracteres en un documento dado.
 - Número de caracteres por oración. Se cuantifica para cada documento el número de caracteres que posee cada oración.
 - Signos de puntuación. Se cuantifican todos los delimitadores de frases y párrafos que establecen una jerarquía sintáctica.

¹ Se utilizó el paquete NLTK para obtener estas colocaciones.

3.2. Método de clasificación

Sea $\langle x_1, x_2, \dots, x_n \rangle$ el conjunto de características seleccionadas para representar los documentos. Cada documento es representado considerando la frecuencia de aparición de cada una de las características presentadas anteriormente, es decir, se usa el modelo de bolsa de palabras ponderadas para representar a cada documento[5].

Como modelo clasificador se utiliza una máquina de soporte vectorial (Support Vector Machine, por sus siglas en inglés SVM). SVM es un sistema de aprendizaje basado en el uso de un espacio de hipótesis de funciones lineales en un espacio de mayor dimensión inducido por un Kernel, en el cual las hipótesis son entrenadas por un algoritmo tomado de la teoría de optimización, el cual utiliza elementos de la teoría generalización. Debido a las limitaciones computacionales de las máquinas de aprendizaje lineal, éstas no pueden ser utilizadas en la mayoría de las aplicaciones del mundo real. La representación por medio del Kernel ofrece una solución alternativa a este problema, proyectando la información a un espacio de características de mayor dimensión, lo cual aumenta la capacidad computacional de las máquinas de aprendizaje lineal. Se mapea el espacio de entrada X a un nuevo espacio de características:

$$x = \{x_1, x_2, \dots, x_n\} \rightarrow \phi(x) = \{\phi(x)_1, \phi(x)_2, \dots, \phi(x)_n\} \quad (1)$$

Utilizando la función Kernel no es necesario calcular explícitamente el mapeo $\phi : X \rightarrow F$ para aprender en el espacio de características.

En el caso de esta investigación se utilizó como Kernel el mapeo polinomial que es un método muy popular para modelar funciones no lineales:

$$K(x, x) = (\langle x, x \rangle + C)^d \quad (2)$$

donde $C \in R$.

La SVM utilizada es la implementada en la plataforma de software para aprendizaje automático y minería de datos WEKA[7], ya que está disponible como software libre, es portable y contiene una extensa colección de técnicas para el preprocesamiento de datos y modelado.

4. Evaluación

Se desarrollaron 18 experimentos para evaluar las características de autoría planteadas anteriormente. En los 17 primeros experimentos se usa siempre una sola característica, y se representa a cada autor considerando la frecuencia ésta en cada uno de sus documentos. El último experimento considera entrenar el modelo de clasificación usando las seis mejores características de acuerdo a las evaluaciones obtenidas en los primeros 17 experimentos. El proceso de entrenamiento se llevó a cabo en los corpora que a continuación se describen:

4.1. Conjunto de datos

Se trabajó con tres conjuntos de datos en inglés, el primero y segundo fueron propuestos en el marco del congreso PAN 2011. Se trata de dos colecciones de textos desbalanceadas conteniendo chats de la Empresa Enron; el primero (denominado *Small*) tiene 26 autores diferentes, y el segundo (llamado *Large*) tiene 72 autores diferentes. El tercer conjunto de datos (llamado *C50*) fue creado por Stamatatos a partir de la clase CCAT perteneciente a la colección RCV1 [4]. C50 consiste de noticias corporativas industriales y es básicamente una colección de documentos balanceados y disjuntos con 50 autores diferentes. En la Tabla 1 se muestra el número de documentos usados tanto para la fase de entrenamiento como para la fase prueba.

Tabla 1. Número de documentos, para entrenamiento y prueba, asociado a cada conjunto de datos

Conjunto de datos	Fase de entrenamiento	Fase de Prueba
C50	2,500	2,500
PAN Small	3,001	518
PAN Large	9,337	1,298

4.2. Resultados obtenidos

Los resultados para cada uno de los primeros 17 experimentos se pueden observar en la Tabla 3. Entre las características que obtienen un mejor rendimiento, es decir, que lograron recuperar mejor los rasgos distintivos de cada autor se encuentran las siguientes (se muestran los resultados para cada corpus):

- Trigramas del texto categorizado (44.92 %, 30.97 % y 58.30 %)
- Palabras cerradas (29.96 %, 18.48 % y 64.47 %)
- Permutación de las vocales (37.96 %, 7.70 % y 83.59 %)
- Sufijos de palabras (33.80 %, 7.30 % y 42.08 %)
- Combinación de las vocales (24.64 %, 6.50 % y 77.41 %)
- Letras del abecedario (28.88 %, 15.17 % y 71.62 %)

Sin embargo, la selección de las mismas contempló no solamente el porcentaje de valores correctos, sino también el cubrimiento de los documentos que corresponden a los autores. El proceso de selección se llevó a cabo de la siguiente manera. Primeramente se eligió la característica que obtuvo el mejor rendimiento global en las tres colecciones (en este caso, el “trigrama de texto categorizado”). Posteriormente, se fueron agregando aquellas características con buen porcentaje de documentos clasificados correctamente, pero que además recuperan documentos que las características anteriores no habían recuperado.

Con estos resultados en mente, se decide entonces realizar un nuevo experimento considerando ahora la frecuencia de las seis características anteriormente comentadas. Los resultados obtenidos pueden verse en la Tabla 3.

Tabla 2. Resultados de la representación de las características individuales

Característica	C50 (2500 docs) % (correctos)	PAN Large (1298 docs) % (correctos)	PAN Small (518 docs) % (correctos)
Palabras de frecuencia 1	11.16 % (279)	13.32 % (173)	24.90 % (129)
Palabras de frecuencia 2	11.80 % (295)	10.86 % (141)	20.07 % (104)
10 palabras más frecuentes	16.48 % (412)	10.86 % (141)	20.07 % (104)
Colocaciones	12.92 % (323)	17.48 % (227)	51.93 % (269)
Signos de puntuación	16.32 % (408)	16.79 % (218)	58.88 % (305)
Combinación de las vocales	24.64 % (616)	6.50 % (85)	77.41 % (401)
Permutación de las vocales	37.96 % (949)	7.70 % (100)	83.59 % (433)
Sufijos de las palabras	33.80 % (845)	7.30 % (96)	42.08 % (218)
Prefijos de las palabras	15.80 % (395)	15.79 % (205)	59.65 % (309)
Trigrama de palabras	17.00 % (425)	23.11 % (300)	40.34 % (209)
Trigrama de caracteres	12.04 % (301)	24.73 % (321)	32.81 % (170)
Trigrama de texto categorizado	44.92 % (1123)	30.97 % (402)	58.30 % (302)
Núm. de oraciones	13.80 % (345)	14.71 % (191)	30.69 % (159)
Núm. palabras por oración	15.16 % (379)	15.94 % (207)	15.05 % (78)
Núm. caracteres por oración	16.08 % (402)	14.40 % (187)	41.11 % (213)
Palabras cerradas	29.96 % (749)	18.48 % (240)	64.47 % (334)
Letras del abecedario	28.88 % (722)	15.17 % (197)	71.62 % (371)

El número de documentos recuperados correctamente por el modelo de clasificación que conjuga las seis características es significativamente mejor que cualquiera de los resultados obtenidos de manera individual. Es interesante observar que en el caso del corpus *Small* del PAN se obtiene un porcentaje muy alto, lo cual podría ser derivado del hecho de que este corpus contiene documentos con una longitud mucho mayor que los otros dos corpora. Esto llevaría a poder identificar más fácilmente ciertos patrones de escritura del autor.

Si bien, los resultados para los corpora *C50* y *PAN Large* aun se encuentran alrededor del 50 %, consideramos que es posible mejorarlos. Una línea de investigación que estamos trabajando es usando representaciones alternativas, como por ejemplos los grafos, y usando algoritmos que reconocen automáticamente, ciertos patrones, en este caso, de escritura.

Tabla 3. Resultados de la combinación de características

Característica	C50 (2500 docs) % (correctos)	PAN Large (1298 docs) % (correctos)	PAN Small (518 docs) % (correctos)
6 Mejores características	53.20 % (1330)	46.53 % (604)	91.31 % (473)

5. Conclusión

En este trabajo de investigación se reporta una evaluación de diferentes características usadas para el proceso de atribución de autoría. Se puede observar, que para uno de los corpora es posible obtener un porcentaje alto de documentos clasificados correctamente, lo cual apoya la teoría de la posibilidad de detectar los rasgos distintivos de un determinado estilo de escritura. Para el caso de los otros dos corpora, con las características estudiadas se ha logrado clasificar correctamente más del 53 % de los documentos en el conjunto de datos *C50*, y

el 46% de los documentos para el conjunto de datos *PAN Large*. Esto indica que se debe hacer un estudio más detallado para capturar patrones de escritura distintivos en el caso de textos cortos.

Como trabajo a futuro, nos planteamos analizar si modificando la representación textual mediante grafos, es posible mejorar los resultados de clasificación.

Referencias

1. Argamon, S., Juola, P.: Overview of the international authorship identification competition at PAN-2011. In: Petras, V., Forner, P., Clough, P.D. (eds.) CLEF (Notebook Papers/Labs/Workshop) (2011)
2. Juola, P.: Authorship attribution. *Found. Trends Inf. Retr.* 1(3), 233–334 (Dec 2006), <http://dx.doi.org/10.1561/1500000005>
3. Kern, R., Seifert, C., Zechner, M., Granitzer, M.: Vote/Veto Meta-Classifier for Authorship Identification - Notebook for PAN at CLEF 2011. In: Petras, V., Forner, P., Clough, P.D. (eds.) CLEF (Notebook Papers/Labs/Workshop) (2011)
4. Lewis, D.D., Yang, Y., Rose, T.G., Li, F.: Rcv1: A new benchmark collection for text categorization research. *J. Mach. Learn. Res.* 5, 361–397 (Dec 2004), <http://dl.acm.org/citation.cfm?id=1005332.1005345>
5. Manning, C., Raghavan, P., Schtze, H.: Introduction to information retrieval. In: Same working notes. pp. 117–120 (2008)
6. Stamatatos, E.: A survey of modern authorship attribution methods. *J. Am. Soc. Inf. Technol.* 60(3), 538–556 (Mar 2009), <http://dx.doi.org/10.1002/asi.v60:3>
7. Witten, I.H., Frank, E.: *Data Mining: Practical Machine Learning Tools and Techniques*. The Morgan Kaufmann Series in Data Management Systems, Morgan Kaufmann Publishers, San Francisco, CA, 2nd edn. (2005)

Análisis de una aplicación multilingüe del agrupamiento de textos

Alisa Zhila y Alexander Gelbukh

Centro de Investigación en Computación,
Instituto Politécnico Nacional, México, D.F.
alisa.zhila@gmail.com, www.gelbukh.com

Resumen Muchas tareas del procesamiento de lenguaje natural, tales como la traducción automática, la desambiguación de los sentidos de las palabras, y la detección de textos traducidos, entre otras, requieren del análisis de la información contextual. En el caso de los enfoques del aprendizaje automático supervisado, este análisis se debe previamente llevar a cabo por expertos humanos, lo cual es prohibitivamente costoso. Los enfoques del aprendizaje automático no supervisado ofrecen métodos totalmente automáticos para la resolución de las mismas tareas. Sin embargo, estos métodos no son robustos; sus resultados dependen mucho de los parámetros elegidos y son difíciles de interpretar. El agrupamiento de textos es una técnica no supervisada para el análisis de similitudes entre los textos. En este trabajo exploramos la utilidad del agrupamiento de textos para la detección de traducción de una palabra. Específicamente, analizamos cómo los resultados del agrupamiento dependen de los parámetros del agrupamiento y evaluamos los resultados comparándolos con las traducciones hechas por traductores humanos.

Palabras clave: Agrupamiento de textos, traducción automática, desambiguación en la traducción, métodos no supervisados, selección de parámetros, contexto.

1. Introducción

En el procesamiento de lenguaje natural [16] la tarea de la traducción automática es uno de los problemas más importantes de este campo. La tarea de detección de traducción de una palabra se considera como un caso particular de la traducción automática. Esta tarea consiste en la selección de la mejor traducción, o varias traducciones equivalentes (sinónimas), para una palabra polisémica en un contexto dado, de un conjunto de variantes de traducción ofrecido por el diccionario bilingüe.

En la última década fueron explorados los enfoques para el mejoramiento de la traducción automática con la desambiguación de los sentidos de las palabras efectuada antes de la traducción [1, 14]. Estos enfoques se basan en los clasificadores supervisados, los cuales requieren un amplio entrenamiento con un corpus etiquetado manualmente. Requieren de muchos recursos computacionales pero proporcionan una mejora relativamente poca a un costo alto.

La tarea de la desambiguación de traducción, también llamada detección de la traducción, fue considerada en [4, 15] con recursos costosos, ya sea con un clasificador supervisado o con un corpus monolingüe etiquetado muy grande.

El agrupamiento de textos es un enfoque no supervisado para la detección de similitud entre los textos [6, 10]. Sin embargo, sus resultados dependen en gran medida de los parámetros utilizados para el agrupamiento. Este enfoque se aplicó a la tarea de discriminación de los sentidos de las palabras [9], la cual consiste en detección de la existencia de diferentes acepciones de la palabra sin que éstas estén previamente especificadas en algún diccionario.

La aplicación de los enfoques no supervisados a la detección de la traducción de una palabra no se ha explorado todavía. Nuestra hipótesis es que el agrupamiento no supervisado de textos aplicado a un corpus de textos paralelos alineados por palabras puede ser útil para obtener las características del contexto que permitan la selección correcta de una variante de traducción para una palabra en un contexto dado de manera no supervisada.

En este trabajo exploramos la utilidad del agrupamiento de textos para la detección de la traducción mediante la comparación de los resultados del agrupamiento obtenidos con varias combinaciones de los parámetros. Evaluamos los resultados obtenidos comparándolos con las traducciones humanas obtenidas del corpus de textos paralelos alineados.

El resto del artículo está estructurado de la siguiente manera. En la sección 2 se explican los conceptos del agrupamiento de texto. En la sección 3 se dan a conocer los parámetros de los algoritmos que hemos explorado en este trabajo. En la sección 4 se presentan los resultados obtenidos. Finalmente, la sección 5 presenta una discusión de las lecciones aprendidas y las conclusiones.

2. El agrupamiento de los textos

En la última década el tema de la discriminación no supervisada de los sentidos de las palabras (WSD por sus siglas en inglés: *word sense discrimination*), o sea, la distinción entre los usos diferentes de una palabra en contextos diferentes, se ha investigado activamente. La solución más conocida a este problema es el agrupamiento de los contextos que contienen la palabra en cuestión [6, 10]. Una revisión amplia del agrupamiento como la clasificación no supervisada de los elementos del conjunto de datos en grupos se presenta en [2]. Los algoritmos de agrupamiento adecuados para el agrupamiento de documentos se describen y analizan en [11]. Estos algoritmos son implementados en la herramienta de agrupamiento que se llama SenseClusters [8]. Nosotros adoptamos este programa como la herramienta central para nuestros experimentos.

Sin embargo, los resultados del agrupamiento de textos dependen en gran medida de los parámetros con los cuales se realiza el agrupamiento. En esta sección presentamos una breve descripción de los parámetros y las técnicas del agrupamiento.

2.1. Los tipos de las características de los textos

Para llevar a cabo un agrupamiento de elementos, antes de todo es necesario elegir características para representar cada elemento. En el campo del agrupamiento automático de los textos, las características son usualmente secuencias de palabras,

llamadas *n*-gramas. Los *unigramas* son las características que se forman de una sola palabra. Pares de palabras consecutivas se denominan *bigramas*. Una definición extendida en [3] propone que bigramas son pares de palabras que se encuentran dentro de cierta distancia (*ventana*) una de otra, preservando el mismo orden en el cual aparecen en el texto. Los pares de palabras dentro de una ventana sin importancia del orden se llaman *coocurrencias*. Para los textos que contienen una palabra específica cuyas propiedades están bajo el estudio, como en el caso de la discriminación de los sentidos de la palabra dada, se consideran *coocurrencias con la palabra objetivo* (es decir, la palabra dada), las cuales son las coocurrencias que contienen la palabra en cuestión.

Las características que se encuentran un número de veces menor de un *umbral* r (el parámetro de corte de la frecuencia) no pueden servir como una base sólida para el agrupamiento de textos y, por lo tanto, se excluyen de la lista de las características.

2.2. El orden de la representación de los textos

Se consideran representaciones de primer y de segundo orden. La representación de primer orden representa un texto como un vector solamente de las características que se encuentran directamente en el texto. La representación de segundo orden considera también las características que se encuentran junto con las características del texto dado, en otros textos existentes.

Por ejemplo, si el texto A es “ratón de computadora”, sus unigramas de primer orden son “ratón” y “computadora”. La preposición “de” se elimina ya que aparece en una lista de palabras basura (*stopwords* en inglés), las cuales son las palabras que no agregan información específica sobre el texto. En el caso de la presentación de segundo orden, considerando también el texto B “ratón inalámbrico”, las características del texto A resultan ser “ratón”, “computadora” e “inalámbrico”. De ese modo se aumenta la cantidad de las características en casos de los textos cortos.

2.3. Medidas de similitud

Para evaluar la similitud entre los textos, es necesario introducir una medida de similitud correspondiente a la representación seleccionada de las características. Usualmente los elementos son representados como vectores de características, con lo cual tales se pueden utilizar tales medidas de similitud entre los vectores como distancia o coseno. Un conjunto de textos se puede representar en un espacio vectorial, donde un vector corresponde a un documento, o bien se puede construir una matriz de similitudes entre pares de textos.

2.4. Funciones de criterio de agrupamiento

La tarea del agrupamiento de elementos es la optimización de una función llamada la función de criterio de agrupamiento (*clustering criterion function* en inglés), la cual es una función de medida de similitud. Una revisión y comparación de las funciones de criterio de agrupamiento se presenta en [6], donde los autores comparan varias funciones de criterio de agrupamiento. Las funciones de criterio internas denominadas

I1, I2 e I3 [6] se basan en la similitud dentro de un grupo de elementos, mientras que las funciones de criterio externas denominadas E1 y E2 toman en cuenta las distancias entre los grupos. Las funciones denominadas H1 y H2 son híbridas: combinan las propiedades de las funciones de criterio internas y externas. Todas estas funciones consideran un documento o texto como un vector de características. En [6] se muestra que las funciones I2 y H2 dan los mejores resultados con la mayoría de los algoritmos de agrupamiento.

2.5. Técnicas de agrupamiento

Existen diversas técnicas, o algoritmos, de agrupamiento. Éstas se dividen en dos grandes categorías: jerárquicas y de partición. La descripción detallada de las técnicas de agrupamiento y la comparación de sus aplicaciones para el agrupamiento de textos se presenta en [11]. Esa investigación mostró que las mejores técnicas de agrupamiento de textos son las siguientes: *k*-medias (*k-means* en inglés), la cual es una técnica de partición con *k* centros de agrupamiento, y el agrupamiento aglomerativo con *UPGMA* (por sus siglas en inglés, *Unweighted Pair Group Method with Arithmetic mean*), el cual es un algoritmo jerárquico.

2.6. Criterios de paro del agrupamiento

Las técnicas de agrupamiento existentes requieren que el número de los clústeres resultantes sea conocido de antemano, lo cual no se cumple en muchos casos prácticos, especialmente para el agrupamiento de textos y la discriminación de los sentidos de las palabras. Para los casos cuando el número deseado de grupos no está conocido de antemano, en [3] se introducen los criterios de paro automático del proceso de agrupamiento. El criterio *gap* se basa en la llamada estadística GAP aplicada a la dispersión de los elementos dentro de cada grupo. Los criterios PK1, PK2 y PK3 (PK refiere a “Predecir el número *K* de grupos”) miden el valor del cambio de la función de similitud de cada grupo con el número *k* de los grupos.

En este trabajo se utilizan los criterios de paro de agrupamiento para detectar automáticamente el número de sentidos del término en cuestión, ya que la idea de la discriminación de sentidos de palabra sostiene que los usos similares de una palabra se encuentran en los contextos similares. A continuación, comparamos el número resultante de los grupos con el número de acepciones proporcionadas en los diccionarios, para evaluar el desempeño de los criterios de paro automático del agrupamiento.

2.7. Evaluación del agrupamiento

La evaluación del desempeño de los algoritmos de agrupamiento depende considerablemente de la medida que se utiliza para evaluarlo [11]. Hay dos enfoques básicos para la evaluación del agrupamiento: interno y externo. Las medidas internas no utilizan ningún conocimiento externo sobre las posibles soluciones (clases de elementos). Las medidas externas comparan los resultados de agrupamiento contra las clases de los elementos conocidas con algún procedimiento externo.

Nosotros evaluamos los resultados del agrupamiento comparándolos con el conjunto de las variantes de traducción obtenidos con un corpus de textos paralelos, véase la sección 3. Elegimos las medidas externas de la entropía y la pureza, cuyas definiciones y fórmulas las adoptamos del trabajo [11].

La entropía es la medida de la incertidumbre de la distribución de las clases por los grupos. Para un grupo S_r con n_r elementos, la entropía del grupo es

$$E(S_r) = -\frac{1}{\log q} \sum_{i=1}^q \frac{n_r^i}{n_r} \log \frac{n_r^i}{n_r},$$

donde q es el número de las clases y n_r^i es el número de los elementos de la clase i que se encuentran en el grupo r . La entropía resultante es la sumatoria ponderada de las entropías de todos los clústeres:

$$E(S_r) = \sum_{r=1}^k \frac{n_r}{n} E(S_r), \quad (1)$$

donde k es el número de los grupos y n es el número de todos los elementos en el conjunto de datos.

La pureza de un cluster se calcula como:

$$P(S_r) = \frac{1}{n_r} \max_i(n_r^i)$$

Es la fracción más grande de un grupo formada por los elementos de una clase, es decir, es la fracción de la clase más grande en un grupo.

La pureza total de un conjunto de grupos es la sumatoria ponderada de todas las purzas individuales de los clústeres en el conjunto:

$$P = \sum_{r=1}^k \frac{n_r}{n} P(S_r).$$

La pureza evalúa la calidad con la cual cada clase corresponde a un grupo. En resumen, la entropía más baja y la pureza más alta indican mejor calidad del agrupamiento.

3. Metodología experimental

En esta sección describimos los experimentos llevados a cabo en este trabajo. Nuestro objetivo es la exploración de la utilidad del agrupamiento no supervisado de textos para la detección de la traducción. Para este experimento hemos utilizado la herramienta existente SenseCluster [8] para realizar el agrupamiento de los textos de nuestra base de textos.

Nuestros experimentos se basan en los resultados obtenidos en [7, 9, 11, 13]. Exploramos si estos parámetros de agrupamiento son apropiados para la detección de la traducción. Nuestra hipótesis principal es que las traducciones son correlacionadas con las acepciones de la palabra que se traduce.

La discriminación no supervisada de los sentidos de las palabras sostiene que los usos similares con las mismas acepciones se agrupan. Por lo tanto, los contextos de un grupo corresponderían a una sola traducción o a las traducciones sinónimas. Utilizamos los criterios de paro del agrupamiento para detectar automáticamente el número de acepciones de la palabra en cuestión.

A continuación, usamos el número de acepciones proporcionadas por los diccionarios como el umbral para el número de los grupos.

3.1. El corpus de textos

Extrajimos nuestro corpus de textos para el experimento del corpus paralelo inglés-español Europarl disponible con el “corpus abierto OPUS” [12].

Para nuestro propósito de explorar la posibilidad del uso del agrupamiento de textos para la detección de la traducción, una palabra ambigua tenía que satisfacer los siguientes criterios:

- ser encontrada en el corpus paralelo al menos mil veces, lo cual es suficiente para llevar a cabo el agrupamiento de textos no supervisado;
- tener más de una variante de traducción en la parte paralela del corpus.

Hemos encontrado las siguientes palabras en la parte inglesa del corpus que satisfacían estos dos criterios: *facility*, *post*, *language*. Debido a las limitaciones del espacio, en este artículo sólo presentamos los resultados para la palabra *facility*.

Para formar un contexto extrajimos 7 oraciones consecutivas del corpus de manera tal que una de ellas contenga la palabra elegida. Elegimos el tamaño de 7 oraciones basándonos en los tamaños promedios de los contextos utilizados en [7, 9, 11, 13]. En este paso, extrajimos 1771 contextos.

Luego eliminamos los contextos mal alineados de nuestro conjunto experimental de textos. Los casos en los cuales la palabra *facility* no tenía un equivalente de traducción los marcamos como NOTAG (palabra no etiquetada). Agrupamos las variantes de las traducciones de baja frecuencia (de 1 a 6 ocurrencias en el corpus) con sus sinónimos, prestando atención a sus usos en contexto. Finalmente, obtuvimos un conjunto de 1429 contextos y 21 clases de traducción, incluyendo la clase NOTAG¹. Estas clases de traducción sirven para la evaluación externa de los clústeres resultantes.

Los diccionarios monolingües que hemos consultado (Merriam-Webster en línea, *Oxford Concise Thesaurus*, WordNet y *Larousse American Pocket*) presentan entre 4 y 5 acepciones para la palabra *facility*. Nosotros adoptamos el menor de estos números (4) como el umbral para el número mínimo de los grupos. Por eso,

¹ El conjunto de los textos está disponible en <http://www.gelbukh.com/resources/word-translation-alignments/>

eliminamos cualquier combinación de valores de parámetros que genera menos de 4 grupos.

3.2. Los parámetros del agrupamiento

En este trabajo llevamos a cabo el agrupamiento de textos con la herramienta SenseClusters [8]. Es un sistema para el agrupamiento de textos completo y libremente disponible, el cual proporciona las posibilidades para la selección de las características de los textos, varios esquemas de representación de éstos, los algoritmos de agrupamiento diferentes y la evaluación de los grupos obtenidos.

Los parámetros con valores fijos. Los valores de los siguientes parámetros no se variaron en nuestros experimentos:

- el orden de la representación: usamos la representación de segundo orden (O_2);
- los textos fueron representados como vectores de características en el espacio vectorial;
- el ancho de la ventana entre las palabras de características fue 5;
- el parámetro r de la frecuencia de corte fue 3.

Elegimos la representación de los contextos de segundo orden ya que se ha demostrado que esta representación es mejor para los contextos cortos [5]. El espacio vectorial es preferible a la matriz de similitud según el trabajo [9]. Para el valor del ancho de la ventana tomamos como referencia el trabajo [7]. Hemos establecido el valor del parámetro r de la frecuencia de corte a 3 heurísticamente, basándonos también en el trabajo [7].

Los parámetros con los cuales experimentamos. Ya que el número total de las posibles combinaciones de los valores de los parámetros es muy alto, analizamos sólo los valores de los parámetros que se han demostrado ser mejores para el agrupamiento de textos [11, 13]. Los parámetros con los valores que variamos eran los siguientes:

- las características para la representación de contexto: unigramas, bigramas, coocurrencias y coocurrencias con la palabra objetiva;
- las técnicas del agrupamiento: k-medias, bisección repetida, bisección repetida refinada y el agrupamiento aglomerativo;
- las funciones de criterio de agrupamiento: I_2 , H_2 y UPGMA;
- los criterios de paro del agrupamiento: gap , PK1, PK2, PK3.

El número total de los experimentos era 112.

4. Resultados experimentales

El número de los grupos que obtuvimos con diversas combinaciones de los parámetros del agrupamiento variaba de 1 a 6.

Los criterios de paro del agrupamiento. La tabla 1 muestra las frecuencias de cada número de los grupos para los criterios de paro del agrupamiento.

Tabla 1. Número de grupos obtenido con cada criterio de paro.

Criterio	Número de grupos					
	debajo del umbral			arriba del umbral		
	1	2	3	4	5	6
<i>gap</i>	24	0	4	0	0	0
PK1	11	10	3	1	3	0
PK2	0	8	10	3	4	3
PK3	0	12	9	6	1	0

Como se observa en la tabla 1, los criterios de paro del agrupamiento *gap* y PK1 resultan en el menor número de grupos. El criterio *gap* no ha producido el número de grupos más del umbral de 4. El criterio PK1 dio resultados aceptables sólo en 4 casos, lo cual representa el 3.5% de los casos.

La proporción del número de experimentos por cada cantidad de los grupos obtenidos se muestra en la tabla 2.

Tabla 2. Proporción del número de experimentos por cantidad de los grupos obtenidos.

número de grupos	1	2	3	4	5	6
proporción de experimentos, %	31.2	26.8	23.2	9.0	7.1	2.7

El 50% de la cantidad total de los experimentos fueron obtenidos de 2 a 3 grupos, y sólo el 18,8% (21 del total de los 112 experimentos) pasaron sobre el umbral de 4 grupos.

La suposición de que la palabra *facility* puede tener sólo 2 a 3 acepciones reales o bien distinguidos no parece sólida. Analizando las definiciones de *facility* en los diccionarios se puede observar que sus acepciones difícilmente se puedan agrupar en la cantidad de acepciones independientes y semánticamente no relacionadas menor de 4 acepciones.

Por lo tanto, interpretamos las cantidades bajas de los grupos que se produjeron con los criterios de paro del agrupamiento *gap* y PK1 como una cualidad inherente a estos criterios. Los criterios PK2 y PK3 dieron resultados aceptables en el 36% y el 25% de los casos, respectivamente.

Características de contextos, la entropía y la pureza. Para cada experimento con el número de clústeres resultantes mayor que el umbral de 4, los valores de los parámetros de la entropía y la pureza correspondientes se presentan en la tabla 3. Las columnas en esta tabla son las siguientes: *técnica* refiere a la técnica de agrupamiento, *criterio* refiere a la función del criterio de agrupamiento, *paro* refiere al criterio de paro del agrupamiento, *grupos* es el número resultante de los grupos para la combinación dada de los parámetros y *E* y *P* son la entropía y la pureza, respectivamente. Las abreviaturas son como sigue: *agglo* para la técnica de agrupamiento aglomerativo, *direct* para k-medias, *br* para bisección repetitiva, *brr* para bisección repetitiva refinada. Estas abreviaturas se utilizan en el resto del artículo. Otros términos se han explicado en las secciones 2 y 3.2.

Ninguno de los experimentos con las bigramas o las coocurrencias con la palabra objetiva resultaron en el número de grupos mayor o igual a 4. Por lo tanto, los

resultados de todos los experimentos con estas características se descartaron y no se muestran en la tabla 3.

Tabla 3. Resultados obtenidos con diferentes técnicas de agrupamiento.

técnica	criterio	paro	grupos	E	P
Coocurrencias					
agglo	upgma	pk2	6	80.6	25.5
direct	h2	pk1	4	80.4	25.6
direct	h2	pk3	4	80.4	25.6
direct	i2	pk2	5	80.2	25.5
direct	i2	pk3	4	80.4	25.6
br	h2	pk1	5	80.7	25.0
br	h2	pk2	4	81.0	25.0
br	h2	pk3	4	81.0	25.0
br	i2	pk3	5	80.7	25.0
brr	h2	pk3	4	80.4	25.6
brr	i2	pk2	5	80.2	25.5
Unigramas					
agglo	upgma	pk2	6	84.1	24.2
direct	i2	pk2	6	74.8	26.9
br	h2	pk1	5	75.2	28.3
br	h2	pk2	4	76.2	27.6
br	h2	pk3	4	76.2	27.6
br	i2	pk2	5	75.6	27.8
brr	h2	pk1	5	75.2	28.3
brr	h2	pk2	4	76.2	27.6
brr	h2	pk3	4	76.2	27.6
brr	i2	pk2	5	75.3	28.3

En la tabla 3 los mejores valores de la entropía y la pureza se dan en negritas. Todos ellos fueron obtenidos para las combinaciones de los parámetros con las unigramas. Generalmente, la entropía y la pureza para las unigramas son aproximadamente 5% y 12% mejores, respectivamente, que las de las coocurrencias. Sin embargo, la comparación de los valores de la entropía y la pureza con los obtenidos en [11, 13] se obstaculiza porque la entropía y la pureza dependen del número de las clases externas.

Podemos deducir que la entropía y la pureza como se describen en [11] podrían no ser apropiados para nuestra tarea. En [11], estas medidas se utilizan para evaluar los resultados de la discriminación de los sentidos de las palabras cuando se supone que cada grupo corresponde a una acepción de la palabra y se supone que el número de los grupos sería igual al número de las acepciones, que son las clases externas usadas para la evaluación. En nuestro caso, es perfectamente aceptable que más de una clase se agrupa en un grupo o que los elementos de una clase se distribuyen entre varios grupos, lo cual es interpretable desde el punto de vista de la traducción.

Número de grupos. Para investigar la influencia del número de los grupos en la entropía y la pureza, llevamos a cabo un experimento con un número de grupos fijo establecido de forma manual a 21, que es el número de las clases de traducción. En este experimento se utilizó una combinación de parámetros de agrupamiento que produjo la pureza más alta. Los resultados se muestran en la tabla 4.

Como se observa, aumentando el número de los grupos más de cuatro veces (de 5 a 21) se mejoran los valores de la entropía y la pureza sólo al 10.6% y 15.5%, respectivamente.

Tabla 4. Los resultados obtenidos con 21 grupos.

técnica	criterio	paro	grupos	E	P
Unigramas con el número de clústeres fijo					
rb	h2	n/a	21	67.2	32.7

Ejemplo de los resultados del agrupamiento. Para ilustrar los resultados del agrupamiento, hemos elegido dos experimentos que dieron los mejores resultados: uno para las coocurrencias y el otro para los unigramas. La tabla 5 muestra la distribución de las clases de traducción por los grupos resultantes para estas dos combinaciones de los parámetros.

En la sección 3 hemos explicado la suposición de que el agrupamiento no supervisado de textos sería apropiado para la detección de la traducción de la palabra si un grupo correspondiera a una o más clases enteras de traducción, lo que es el caso de sinonimia entre las traducciones, o bien si una clase de traducción fuera distribuida entre algunos grupos, lo que es el caso de la homonimia que se presenta en ambos idiomas. En estos casos, la tabla de la distribución de las clases por los grupos se vería más “diagonal”, o bien tendría más celdas con ceros que las con no ceros. Sin embargo, en la tabla 5 observamos que casi todas las celdas tienen valores distintos de cero. Esto significa que un contexto correspondiente a cualquier variante de traducción se puede encontrar en cualquier grupo, lo que en este caso invalida nuestra hipótesis inicial acerca de la idoneidad del agrupamiento de textos para la detección de traducción.

Tabla 5. Ejemplo de la distribución de las clases por los grupos, en dos experimentos.

Experimento	Tamaño del grupo	<i>dispositivo</i>	<i>equipos</i>	<i>facilidad</i>	<i>instalación</i>	NOTAG	<i>servicio</i>	<i>sistema</i>	<i>posibilidad</i>	<i>mecanismo</i>	<i>institución</i>	<i>capacidad</i>	<i>centro</i>	<i>ayuda</i>	<i>medio</i>	<i>crédito</i>	<i>planta</i>	<i>fondo</i>	<i>medida</i>	<i>infraestruct</i>	<i>plan</i>	<i>central</i>
		Coocurrencias, agrupamiento aglomerativo, UPGMA, PK2, E = 0.806, P= 0.255																				
0	989	42	22	121	196	94	96	19	95	82	16	25	18	19	55	18	16	20	8	14	4	9
1	3	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
2	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	350	3	6	24	142	40	40	3	14	6	10	4	9	0	20	1	6	1	5	2	0	14
4	11	0	0	5	1	1	0	0	2	0	0	0	1	0	0	0	0	0	0	1	0	0
5	75	18	1	1	4	3	5	19	4	8	2	0	1	2	1	1	0	5	0	0	0	0
Unigramas, bisecciones repetitivas, H2, PK1, E = 0.758, P = 0.276																						
0	213	27	1	24	8	17	16	6	24	26	1	9	3	7	12	9	1	15	1	3	2	1
1	156	1	0	1	112	10	3	0	1	1	0	0	1	0	5	0	8	0	2	0	0	11
2	282	0	13	42	83	26	35	3	22	4	6	4	8	3	13	1	3	0	2	9	0	5
3	307	32	1	29	24	25	11	25	32	57	4	9	6	7	18	7	3	10	4	0	2	1
4	471	3	14	56	116	61	77	7	36	8	17	7	11	4	28	4	7	1	4	5	0	5

5. Conclusiones y trabajo futuro

En este trabajo hemos realizado una comparación de las combinaciones de los parámetros del agrupamiento de textos y exploramos su utilidad para la detección no supervisada de la traducción correcta de una palabra. Específicamente, presentamos los resultados para la palabra inglesa *facility* y sus traducciones al español.

Sólo el 18,8% de los experimentos resultaron en el número de grupos que sobrepasó el umbral de 4, que es el número mínimo posible de las acepciones de la palabra *facility*. Entre 2 y 3 grupos se detectaron en el 50% de los casos. En el trabajo actual estos resultados no se interpretan desde el punto de vista semántico y simplemente los excluimos de consideración. Sin embargo, el análisis formal de la similitud semántica de los sentidos a través de una ontología o de una jerarquía semántica puede dar una nueva perspectiva a estos resultados.

Hemos detectado que el uso del criterio de paro del agrupamiento *gap* en los experimentos resulta en el número de grupos muy escaso que no se puede interpretar desde el punto de vista semántico. Logramos obtener los números de los grupos que corresponden a la suposición semántica del número de acepciones de la palabra utilizando los criterios PK2 y PK3. Además, no eliminamos de la consideración el criterio PK1, ya que su uso proporciona el 19% de todos los resultados aceptables.

No hemos detectado resultados aceptables para las coocurrencias con la palabra objetiva y los bigramas. Eso podría explicarse por el valor del parámetro del ancho de la ventana inadecuado y por la dispersión de las características.

La evaluación de los resultados a través de la entropía y la pureza dio los números que no son fáciles de interpretar en el marco de la detección de traducción de una palabra, ya que el número de las clases es mucho mayor que el número de los grupos. Por lo tanto, planeamos trabajar en el desarrollo de una medida diferente del desempeño del agrupamiento la cual sería más adecuada para nuestros objetivos.

Agradecimientos. El trabajo presentado en este artículo fue apoyado parcialmente por el Gobierno de México a través, los proyectos CONACYT 50206-H y 122030 (CONACYT-DST India 2011–2014), SIP-IPN 20121823, así como el SNI y el programa PIFI-IPN.

Referencias

1. Carpuat, M., Wu, D.: Improving statistical machine translation using word sense disambiguation. In: Proc. of EMNLP-CoNLL 2007, pp. 61–72 (2007)
2. Jain, A.K., Murty, M.N., Patrick, J. Flynn, P.J.: Data Clustering: A Review. ACM Computing Surveys, vol. 21, pp. 264–323 (1999)
3. Kulkarni, A., Pedersen, T.: Unsupervised Context Discrimination and Automatic Cluster Stopping. MS Thesis, University of Minnesota, UMSI 2006/90 (2006)
4. Marsi, E., Lynam, A., Bungum, L., Gambäck, B.: Word Translation Disambiguation without Parallel Texts. In: Proc. International Workshop on Using Linguistic Information for Hybrid Machine Translation, Barcelona, Spain (2011)
5. Pedersen T.: Computational Approaches to Measuring the Similarity of Short Contexts: A Review of Applications and Methods. University of Minnesota, UMSI 2010/118 (2008)

6. Pedersen, T., Bruce, R.: Distinguishing word senses in untagged text. In: Proc. of the 2nd Conference on Empirical Methods in Natural Language Processing, Providence, RI, pp. 197–207 (1997)
7. Purandare, A.: Unsupervised Word Sense Discrimination By Clustering Similar Contexts. MS Thesis. University of Minnesota. (2004)
8. Purandare, A., Pedersen, T.: SenseClusters – Finding Clusters that Represent Word Senses. In: Proc. of Fifth Annual Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-04), pp. 26-29 (2004)
9. Purandare, A., Pedersen, T.: Word Sense Discrimination by Clustering Contexts in Vector and Similarity Spaces. In: HLT-NAACL 2004 Workshop, CoNLL-2004, pp. 41-48 (2004)
10. Schütze, H.: Automatic Word Sense Discrimination. *Journal of Computational Linguistics*, vol. 24(1), pp. 97-123 (1998)
11. Steinbach, M., Karypis, G., Kumar, V.: A comparison of document clustering techniques. University of Minnesota, Technical Report 00-034 (2000)
12. Tiedemann, J.: News from OPUS – A Collection of Multilingual Parallel Corpora with Tools and Interfaces. *Recent Advances in Natural Language Processing*, vol. V, pp. 237-248 (2009)
13. Zhao, Y., Karypis, G.: Criterion Functions for Document Clustering: Experiments and Analysis. University of Minnesota, Technical Report 01-040 (2001)
14. Vickrey, D., Biewald, L., Teyssier, M., Koller, D.: Word-sense disambiguation for machine translation. In: Proc. of the conference on Human Language Technology and Empirical Methods in Natural Language Processing 2005, pp. 771–778 (2005)
15. Holmqvist, M.: Memory-based learning of word translation. In: Proc. of the 16th Nordic Conference of Computational Linguistics NODALIDA-2007, Tartu, Estonia, pp. 231-234 (2007)
16. Ledeneva, Y., Sidorov, G.: Recent Advances in Computational Linguistics. *Informatica. International Journal of Computing and Informatics*, 34, 3–18 (2010)

Análisis del conjunto de datos PISA 2009 mediante técnicas de agrupamiento

Yuridiana Alemán, David Pinto, and Nahun Loya y Helena Gómez Adorno

Facultad de Ciencias de la Computación,
Benemérita Universidad Autónoma de Puebla, Puebla, México
yuridiana.aleman@gmail.com, dpinto@cs.buap.mx, israel_loya@hotmail.com,
helena.adorno@gmail.com

Resumen PISA es una evaluación internacional promovida por la OCDE con la finalidad de determinar el nivel de conocimientos y habilidades de estudiantes que están por concluir su educación obligatoria (con una edad alrededor de los 15 años). La última prueba aplicada a nivel mundial fue en el año de 2009 y la próxima será en el 2013. De ahí la importancia de analizar las características que han impactado en el hecho de que en México estemos por debajo de la media. En este trabajo, se atiende esta necesidad desde el punto de vista de la inteligencia artificial, usando algoritmos de agrupamiento (*clustering*). Se aplican dos métodos y se evalúan los experimentos usando la medida F-measure, una medida armónica entre *precision* y *recall*. Los valores obtenidos muestran que el conjunto de características seleccionado para representar la colección de datos es adecuado.

1. Introducción

La Organización para la Cooperación y el Desarrollo Económico (OCDE) coordina la aplicación de PISA, una evaluación internacional desarrollada de manera conjunta por todos los países participantes de la OCDE. Esta evaluación es aplicada con el objetivo de determinar si los estudiantes han adquirido los conocimientos y habilidades relevantes para participar activa y plenamente en la sociedad moderna. PISA se aplica cada tres años, evaluando tres áreas de conocimiento: Matemáticas, Ciencias y Lectura, además; en cada periodo se enfatiza una de estas, por lo que se aplican más reactivos del área seleccionada para su análisis. En la última edición (2009), el área de énfasis fue lectura¹, participando 34 países de la OCDE y 31 asociados [1].

La evaluación de competencias es una parte integral de PISA, y se hace hincapié en el dominio de los procesos, comprensión de conceptos y capacidad para resolver diversas situaciones dentro de cada área [2]. PISA no mide qué tanto pueden reproducir lo que han aprendido, sino que indaga lo que se denomina competencia (*literacy*); es decir, la capacidad de extrapolar lo que se ha aprendido a lo largo de la vida y su aplicación en situaciones del mundo real, así como la

¹ Instituto Nacional para la Evaluación de la Educación: <http://www.inee.edu.mx/>

capacidad de analizar, razonar y comunicar con eficacia al plantear, interpretar y resolver problemas en una amplia variedad de situaciones.

El proceso de evaluación pretende constituirse en la base para la investigación y análisis destinados a mejores políticas en el campo de la educación [3]. PISA ofrece un conjunto de datos del cual se puede obtener información importante, que puede ser procesada por diferentes técnicas (estadísticas, económicas, sociológicas y econométricas). Dentro del campo de la minería de datos se pueden aplicar varias técnicas para el análisis de conjuntos de datos. Un ejemplo sería el uso de técnicas de clasificación con la finalidad de descubrir patrones, reglas de predicción, etc. En este artículo, se utilizan técnicas de clasificación para el análisis de PISA, específicamente clasificación no supervisada.

El objetivo de esta investigación es agrupar a los estudiantes de acuerdo a las respuestas de su examen. Para este propósito se analizan los datos usando dos métodos de clustering: simple k-means y el algoritmo basado en densidad. Se compararán los resultados con el nivel obtenido por los alumnos, a fin de determinar relaciones entre los atributos o características seleccionadas.

El trabajo está estructurado de la siguiente manera: en la Sección 2 se analizan los trabajos relacionados al tema de clasificación y análisis estadístico en el examen PISA. En la Sección 3 se describe la prueba PISA en México, usada en los experimentos de este trabajo. En la Sección 4 se presenta el conjunto de datos usado para llevar a cabo los experimentos de clustering, los métodos de clustering usados, la descripción de las variables y el análisis de los resultados obtenidos. Finalmente se presentan las conclusiones y trabajo futuro.

2. Trabajo relacionado

Algunos autores como [4], [5], [6] y [7] analizan los resultados de la prueba PISA con distintas características, tales como zonas geográficas, atributos cualitativos de estudiantes, entre otros, a fin de proponer cambios en las políticas educativas de algún país o descripción de alguna zona geográfica determinada. Estos estudios son en su mayoría de carácter sociológico o psicológico, y su metodología es principalmente análisis estadístico. En [8], por ejemplo, se analizan los factores determinantes de la calidad educativa para Argentina, especialmente, el papel del tipo de gestión escolar. Para este propósito se utiliza un modelo de regresión multinivel y datos del Programa PISA del año 2006. Entre los principales resultados resalta que la correlación entre el tipo de gestión escolar (pública o privada) y el rendimiento educativo se disipa al considerar el entorno socioeconómico escolar. En ese mismo país, periódicamente la OCDE publica un reporte oficial con todas las estadísticas obtenidas en el procesamiento de los datos [9]. En [1] se presenta una descripción detallada de todo el proceso y recopilación de las muestras de cada edición PISA, añadiendo también el análisis de los resultados en cada área, para obtener el ranking de los países. Cabe notar que las variaciones entre edición y edición no cambian drásticamente, estando los países más desarrollados a la cabeza en todas las áreas. En estas mismas publicaciones, se añade el análisis estadístico de cada país destacando las tendencias

de sus resultados, así como comparaciones internacionales con algunas naciones con Producto Interno Bruto (PIB) similar.

3. PISA 2009 en México

En México, la institución responsable de coordinar la aplicación de la prueba PISA en todas sus fases desde 2003 es el Instituto Nacional para la Evaluación de la Educación (INEE). Este instituto realiza su propio análisis sobre México, y presenta comparaciones de los resultados nacionales con un conjunto de países seleccionados *ex profeso*, como el caso de los iberoamericanos y, sobre todo, centra su atención en el análisis al interior del país a fin de indagar sobre los avances nacionales logrados y los alcanzados por las entidades federativas [10]

En su última edición para México, PISA fue aplicado a una muestra representativa de 38,250 estudiantes (1,535 escuelas). Las preguntas de los exámenes son diseñadas para evaluar el nivel de competencia en cada área. El comité de la OCDE define las competencias como sigue:

Competencia lectora: Capacidad de comprender, utilizar y reflexionar sobre textos escritos, con el propósito de alcanzar sus objetivos personales, desarrollar su conocimiento y sus capacidades, para participar la sociedad [2].

Competencia matemática: Es un concepto que excede al mero conocimiento de la terminología y las operaciones matemáticas, e implica la capacidad de utilizar el razonamiento matemático en la solución de problemas de la vida cotidiana [3].

Competencia científica: Incluye los conocimientos científicos y el uso que de esos conocimientos haga un individuo para identificar preguntas, adquirir nuevos conocimientos, explicar los fenómenos científicos y sacar conclusiones basadas en evidencias, sobre asuntos relacionados con la ciencia [2].

Cada pregunta en el examen, tiene asignado un nivel de dificultad, el cual es calculado a partir del número de alumnos que respondieron bien el reactivo. Una vez calculado el puntaje final, se clasifica al alumno en un nivel de competencia por área. El puntaje mínimo para alcanzar cierto nivel se muestra en la Tabla 1. En el nivel 6, 5 y 4 el estudiante se encuentra en uno de los niveles más altos, por lo que tiene potencial para realizar actividades de alta complejidad cognitiva, científicas u otras; un nivel 3 es bastante bueno, aunque no del nivel deseable para la realización de las actividades cognitivas más complejas. El nivel 2 identifica el mínimo adecuado para desempeñarse en la sociedad contemporánea; por último, un nivel 1 ó 0 es insuficiente para acceder a estudios superiores y desarrollar las actividades que exige la vida en la sociedad del conocimiento.

Además del examen, se aplican dos cuestionarios extra, uno dedicado a los estudiantes, y otro a los directores de las escuelas, en donde se abarcan aspectos culturales, socioeconómicos y académicos. En el 2009, México, se posicionó por debajo de la media establecida por la OCDE, la Tabla 2 se da una referencia de la posición de México en el ranking internacional, con respecto a la media establecida por la OCDE, y los países con mejor y peor puntuación.

Tabla 1. Clasificación por niveles de acuerdo a la OCDE

Nivel	Ciencias	Lectura	Matemáticas
6	707.93	669.3	698.32
5	633.33	606.99	625.61
4	558.73	544.68	552.89
3	484.14	482.38	480.18
2	409.54	420.07	407.47
1	334.94	357.77	334.73
0	-	-	-

Tabla 2. Países y puntajes de PISA 2009

País	Lectura	Matemáticas	Ciencias
Shanghai-China	556	600	575
Promedio OCDE	493	496	501
México	425	419	416
Kyrgyzstan	314	331	330

4. Experimentos realizados

En esta sección se presentan los experimentos de clustering realizados con los datos obtenidos de la prueba PISA 2009. Primeramente se describen los datos usados, así como las técnicas de procesamiento aplicadas. Posteriormente se describen los algoritmos de agrupamiento usados en los experimentos. La medida de evaluación de clusters es mostrada a continuación. Finalmente, se presentan y discuten los resultados obtenidos.

4.1. Preprocesamiento de los datos

Se trabajó con 3 bases de datos obtenidas del sitio oficial de la INEE²: Cuestionario escolar, cuestionario de estudiante y respuestas de los estudiantes. Dada la cantidad de reactivos repartidos entre los diferentes exámenes, el número de atributos de esta base de datos es muy grande, y muy pocos tienen una respuesta en todas las tuplas. De estas bases de datos se utilizaron gráficas y tablas de contingencia para obtener una caracterización general de los datos, con respecto a las escuelas por ejemplo, las privadas son una minoría en todas las modalidades participantes (Secundaria General, Secundaria Técnica, Telesecundaria, Bachillerato General, Bachillerato Tecnológico y Profesional Técnico).

Como el análisis se centra en los estudiantes, se omite la base de datos de las escuelas dentro de las siguientes fases de preprocesamiento:

1. **Sustituir valores perdidos:** Esto se aplicó a algunos atributos, especialmente los referentes a las respuestas de cada una de las preguntas de los

² <http://www.inee.edu.mx/>

exámenes, para obtener los valores de promedio y nivel por área. Algunos atributos se llenaron de manera manual, analizando los elementos de a misma clase. En los demás atributos se trabajó como *valor perdido* (?).

2. **Selección de atributos:** Se eliminaron atributos no discriminantes; como los relacionados al país, algunos atributos de varianzas respecto a la media mundial, y donde los valores perdidos superaban el 50% de las observaciones. En cuanto a las tuplas, no se eliminó ninguna. Además de crear nuevos atributos a partir de los ya existentes. Por ejemplo, las categorías en cada área utilizando las métricas de la OCDE y clasificación del nivel de competencia, además de la obtención de los promedios generales por área, estos muestran un comportamiento estadístico normal (gaussiana) donde la mayoría se agrupa alrededor de los 400 puntos.
3. **Integración:** Se obtuvo una base de datos reducida, a las que se le aplicó una selección de atributos, rescatando algunas variables de carácter socioeconómico, académico y actividades de los alumnos fuera de clases. Las muestras se separaron por categoría (Alto, Bueno, Regular, Insuficiente) para su posterior análisis. Además, se obtuvieron los promedios por estados, tomando en cuenta el promedio general de los alumnos. Los estados con promedios generales más bajos son: Chiapas, Guerrero y Tabasco, mientras que los promedios más altos pertenecen a los estados de Chihuahua, Nuevo León y el Distrito Federal; sin embargo, la diferencia entre el mejor y peor promedio es de solamente 90 puntos.

4.2. Algoritmos de agrupamiento

Para el caso que nos atañe, hemos usado las siguientes dos técnicas de agrupamiento implementadas en la herramienta WEKA³:

Make Density Based Clusters: El cluster se construye basado en las propiedades de densidad de la base de datos y bajo un enfoque de agrupación natural. Los grupos, y en consecuencia las clases, son rápidamente identificables por que tienen una densidad mayor con respecto a los otros puntos [11].

Simple K-means: Asigna cada punto con el grupo cuyo centro (también llamado centro de gravedad) es el más cercano. El centro es el promedio de todos los puntos del cluster, es decir, sus coordenadas son la media aritmética de cada dimensión de todos los puntos en el grupo [11]. Se puede utilizar la distancia euclídeana (por defecto) o la de Manhattan [12].

4.3. Medida de evaluación

Para evaluar la calidad del agrupamiento, se utiliza la medida *F-Measure*, la cual combina de manera armónica los conceptos de *Precisión* y *Recall* de la recuperación de información [13]. Esta métrica requiere por supuesto que los

³ <http://www.cs.waikato.ac.nz/ml/weka/>

corpora usados estén previamente etiquetados. En las ecuaciones 1 y 2 se muestra el cálculo de *Precisión* y *Recall* para el cluster j y la clase i .

$$Pr(i, j) = \frac{n_{ij}}{n_j} \quad (1)$$

$$Re(i, j) = \frac{n_{ij}}{n_i} \quad (2)$$

donde n_{ij} es el número de miembros de la clase i en el cluster j , n_j es el número de miembros del cluster j y n_i es el número de miembros de la clase i .

Los niveles mas altos de *Precisión* generalmente se obtienen con valores bajos de *Recall*. La ecuación 3 permite calcular el valor *F-Measure*, proporcionando un parámetro α ($0 \leq \alpha \leq 1$) que permite ponderar *Precisión* y *Recall*[14].

$$F_\alpha(i, j) = \frac{1}{\alpha \frac{1}{Pr(i, j)} + (1 - \alpha) \frac{1}{Re(i, j)}} \quad (3)$$

Un valor de *F-Measure* es obtenido para toda la colección calculando un promedio pesado de todos los valores de las métricas *F-Measure*, según la ecuación 4.

$$F = \sum_i \frac{n_i}{n} \max_j \{F_\alpha(i, j)\} \quad (4)$$

4.4. Resultados obtenidos

Primeramente, se obtubieron los errores cuadráticos para 2, 3 y 4 clusters, los cuales ascendieron a 352,593.01, 325,685.05 y 311,204.40 respectivamente. Como se observa, los errores son elevados, sin embargo, esto se puede justificar por el número elevado de atributos (33) y la poca correlación que existe en tre estos. De hecho, el menor error se presenta para 4 clusters, pero las muestras se congregan en un sólo grupo.

Con la ayuda de la fórmula F-Measure, se compararon los resultados de los clusters con la clasificación realizada por la INEE, de acuerdo al puntaje obtenido. En la Tabla 3 se comparan los resultados por área y por número de clusters. Éstos indican que la mejor clasificación se obtiene usando sólomente dos clusters (cuando se manejan 4 categorías). Además, el área mejor clasificada fue ciencias, con una F-Measure de 0.67 (K-menas) y 0.73 (Density method). El área de matemáticas es la mas baja en cuanto a *F-Measure* en ambos algoritmos, a pesar de tener una correlación muy fuerte con el área de ciencias. En la Figura 1 se muestra la distribución de los clusters de acuerdo a la categoría en ciencias y del promedio general con el método *Make Density Based Clustered*. Se observa claramente la división de acuerdo a las categorías, donde prácticamente todas las observaciones de nivel *insuficiente* y algunas de nivel *mínimo* son parte del cluster 0, mientras que las categorías de *alto*, *bueno*, y gran porcentaje de *mínimo* son parte de cluster 1.

Tabla 3. Resultados usando F-measure con k-means y cluster basado en densidad

Área	K-means	Densidad
2 Clusters		
Matemáticas	0.613968	0.672767
Ciencias	0.674797	0.732246
Lectura	0.66827	0.716185
3 Clusters		
Matemáticas	0.574676	0.602189
Ciencias	0.622501	0.647574
Lectura	0.589295	0.595035
4 Clusters		
Matemáticas	0.496032	0.494610
Ciencias	0.518151	0.519770
Lectura	0.455953	0.559141

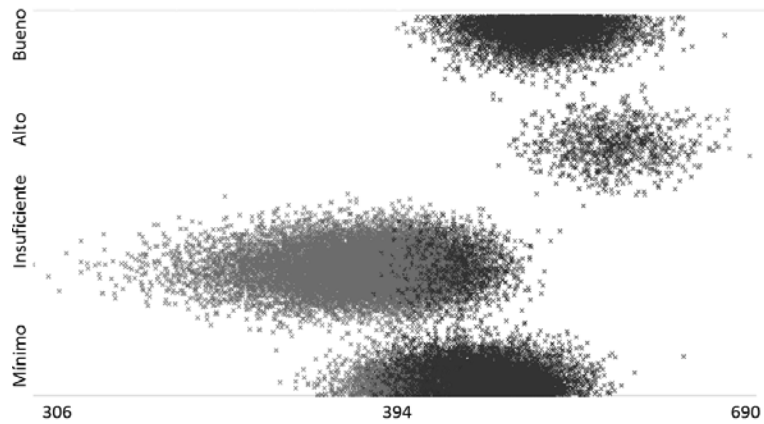


Figura 1. Cluster para categoría de ciencias

Para incrementar el valor de F -measure, de acuerdo a lo presentado en [15], donde se menciona que el análisis de clusters con muchos atributos puede ser difuso si se mezclan atributos de diferente origen, se propone dividirlos y hacer un análisis por separado de acuerdo a cada tópico. Se realizó un análisis con algunas características de los estudiantes como lugar de procedencia, promedios y tipo de escuela. La Tabla 4 muestra los resultados obtenidos. En este experimento sólo se utilizó el algoritmo *Make Density Based Clustered*, ya que en éste se obtuvieron mejores resultados de F -measure. La categoría de ciencias nuevamente obtuvo un mejor puntaje en todos los casos, llegando a 0.8 para dos clusters.

Tabla 4. F-measure para algoritmo por densidad por área

Clusters	Matemáticas	Ciencias	Lectura
2	0.740541	0.803383	0.719292
3	0.610018	0.667056	0.598345
4	0.63634	0.682440	0.606075

Analizando los promedios de los centroides en los clusters obtenidos con el algoritmo *Make Density Based Clustered*, el cluster 0 contiene el promedio general más bajo, y el valor de *insuficiente* en todas las áreas, mientras que el cluster 1 contiene categorías de *bueno* y *regular*, respecto a los demás atributos, los centroides son iguales para los dos clusters, concentrándose en los valores de atributos mas recurrentes, como son el tipo de sostenimiento (público), escolaridad de los padres (básica) y modalidad (bachillerato general).

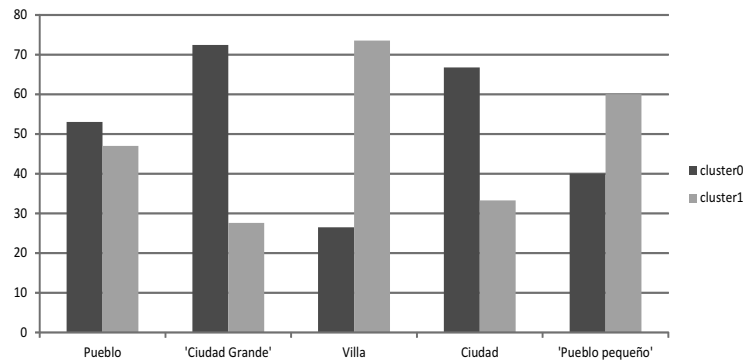


Figura 2. Porcentajes de tipos de localidad distribuidos en los clusters

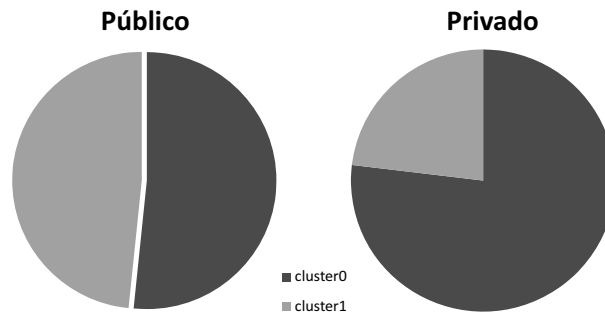


Figura 3. Porcentajes de acuerdo al tipo de sostenimiento de la escuela

En la Figura 2 se muestran los porcentajes de cada tipo de localidad divididas en clusters. En esta Figura se observa que en lugares donde la población es numerosa (*ciudad grande y ciudad*), el porcentaje de estudiantes clasificados en el cluster 0 es superior que en el 1 (en el cluster 0 se congregan los alumnos clasificados en las categorías mas altas), mientras que en lugares con pocos habitantes (*villa, pueblo pequeño*) la situación se invierte, ya que el porcentaje del cluster 1 es mayor, en el caso del valor de Pueblo, es mayor el porcentaje del cluster 0, aunque en una proporción muy pequeña. Dados los promedios de los clusters (tanto general como categorías en cada área), se observa que en las localidades grandes, el nivel de los alumnos es mayor que en pueblos pequeños.

Para analizar esta situación desde otro ángulo, en la Figura 3 se muestran los porcentajes de alumnos asignados a cada cluster de acuerdo al sostenimiento de la escuela (público o privado). Respecto al sostenimiento público, prácticamente la mitad de los alumnos es asignado a cada cluster, pero en el ámbito privado, mas de tres cuartas partes de los alumnos son clasificados en el cluster 0, y sólo una minoría en el cluster 1. Retomando los centroides, los alumnos de las escuelas privadas (a pesar de ser minoría) tuvieron un mejor puntaje.

5. Conclusiones

En este trabajo, se presentó un análisis sobre PISA 2009 usando clasificación no supervisada. Los algoritmos utilizados fueron *Simple K-means* y *Density Based Clustered* implementados en WEKA. El algoritmo basado en densidad obtuvo mejores resultados en todos los casos (*respecto a F-measure*), especialmente al comparar el área de ciencias, donde se logró obtener un 0.80 con una selección manual de atributos. A pesar de que la mayoría de las preguntas del cuestionario de estudiantes fueron sobre lectura, esta área no obtuvo una buena medida respecto a la clasificación por clusters.

Los atributos de PISA 2009 son en su mayoría categóricos, y algunos de ellos binomiales. Con la clasificación no supervisada se pudieron obtener agrupaciones

naturales de los datos, sin restringirlos a una clase determinada. Para este caso, la agrupación en dos clusters fue mejor que en 3 y 4, independientemente del número de niveles por área.

Como trabajo futuro, se llevará a cabo un análisis más profundo y con enfoque pedagógico de los resultados obtenidos a fin de determinar posibles políticas o estrategias para mejorar a los estudiantes del cluster 1 (nivel insuficiente). También se utilizará esta metodología para el análisis de los datos de PISA 2012 México, que se presentarán en el 2013.

Agradecimientos. Este trabajo ha sido apoyado parcialmente por los proyectos CONACYT #106625 y VIEP #PIAD-ING11-II.

Referencias

1. OECD: El programa pisa de la ocde: Qué es y para qué sirve. Technical report, OECD (2006)
2. OECD: Measuring student knowledge and skills: A new framework for assessment. Technical report, OECD (1999)
3. OECD: Assessing scientific, reading and mathematical literacy: A framework for pisa 2006. Technical report, OECD (2006)
4. Cerveró, A.C., i Gràcia, J.V.: La competencia lectora en el estudio pisa. un análisis desde la alfabetización en información. *Anales de Documentación* **8**(8) (2008)
5. Ara, M.J.N., Durán, E.J.U.: Pisa y el triángulo de la evaluación. *Psicothema* **23**(4) (2011) 701–706
6. Rizo, F.M.: Pisa en américa latina: lecciones a partir de la experiencia de méxico de 2000 a 2006. *Revista de Educación* **1**(1) (2006) 153–167
7. Martínez, J.C., Sebastián Waisgrais, A.C.d.M.: Determinantes del riesgo de fracaso escolar en españa: una aproximación a través de un análisis logístico multinivel aplicado a pisa-2006. *Revista de Educación* **1**(1) (2010) 225–256
8. Formichella, M.M.: Se debe el mayor rendimiento de las escuelas de gestión privada en la argentina al tipo de administración? *CEPAL* (105) (2011) 151–166
9. OECD: Pisa 2006 technical report. Technical Report 56393, OECD (2009)
10. Gutiérrez, M.A.D., Vázquez, G.F.: México en pisa 2009. Technical report, INEE (2010)
11. P.Santhi, Bhaskaran, V.: Performance of clustering algorithms in healthcare database. *International Journal for Advances in Computer Science* **2** (2010) 26–31
12. Arthur, D., Vassilvitskii, S.: k-means++: The advantages of careful seeding. Technical Report 2006-13, Stanford InfoLab (June 2006)
13. Frakes, W.B., Baeza-Yates, R.A., eds.: *Information Retrieval: Data Structures & Algorithms*. Prentice-Hall (1992)
14. Arco, L., Bello, R., Mederos, J.M., Pérez, Y.: Agrupamiento de documentos textuales mediante métodos concatenados. *Revista Iberoamericana de Inteligencia Artificial* **10**(30) (2006) 43–53
15. Santana, O.F.: El análisis de cluster : aplicación, interpretación y validación. *Papers : revista de sociologia* (37) (1991) 65–76

Algoritmos bioinspirados

Distribución de puntos de red inalámbricos utilizando algoritmos genéticos

M. Angélica Pinninghoff J., Ricardo Contreras A. y Rodrigo Aravena A.

Departamento de Ciencias de Computación,
Universidad de Concepción, Chile
{rcontrer,mpinning}@udec.cl

Resumen El suministrar servicios de Internet a un determinado lugar, es una tarea cuya dificultad aumenta a medida que el número de equipos que lo usarán y el área que se abastecerá también lo hacen. Solucionar el problema implica el análisis de factores como la ubicación del conjunto de sensores de red, además de las características y restricciones que estos y su ubicación poseen. Así, se trata un problema de optimización que puede ser abordado mediante una categoría de problemas llamada *Facility Location Problem*. En este trabajo se propone aplicar técnicas de algoritmos genéticos para la resolución del problema, obteniendo una distribución de estos sensores para una ciudad arbitraria de manera que se minimicen los costos de instalación sin disminuir la calidad de la señal.

Palabras clave: Algoritmos genéticos, facility location problem, cobertura geográfica.

1. Introducción

Hoy en día las tecnologías forman parte de nuestras vidas, sobre todo las tecnologías que van en apoyo a las comunicaciones. De éstas la principal es Internet, a tal punto que es posible considerarla como uno de los servicios básicos para cada hogar. Las apuestas de personajes influyentes en el área de las tecnologías de la información como Negroponte [9], apuntan a lograr que en la mayoría de los hogares exista a lo menos un computador con conexión a Internet. Por lo anterior, resulta de importancia que una comunidad específica tenga las herramientas necesarias para que todos los integrantes de esa comunidad tengan acceso a esta tecnología.

Ofrecer el acceso a Internet en un determinado lugar geográfico, como por ejemplo el centro de una ciudad específica, debe considerar diversos factores. Lo primero es contar con dispositivos de transmisión que lo permitan, los que se denominan sensores de red. Existen dos formas de realizar esta transmisión; la primera es mediante sensores de red conectados, los cuales requieren de un lugar físico en el cual instalar el cableado y habilitar los puntos de red necesarios para conectarse a Internet. La segunda forma es mediante el uso de sensores de red inalámbricos. Estos sensores se comunican mediante la modulación de ondas electromagnéticas, radiaciones, o medios ópticos. Estas se propagan sin

necesidad de dispositivos físicos, salvo en el emisor y el receptor. Esta forma de conexión posee la ventaja de permitir un acceso a Internet de forma libre y sin necesidad de cables, lo que entrega movilidad y conectividad al usuario.

Para acceder a esta forma de comunicación se debe contar con un computador que disponga de un dispositivo inalámbrico integrado o de lo contrario con un computador compatible con los dispositivos inalámbricos externos existentes en el mercado (USB, PCMCIA, PCI). Dichos dispositivos poseen parámetros de configuración que deben mantenerse dentro de los rangos estándares para obtener un servicio aceptable. Uno de ellos es la intensidad de la señal, la cual se ve afectada directamente por la ubicación de cada uno de estos sensores. Es decir, la intensidad de esa señal depende de la topología del lugar, como también de las características de las construcciones cercanas a la ubicación de cada sensor.

Otro parámetro a considerar es el ancho de banda máximo que posee cada sensor. Éste limita la cantidad de usuarios que pueden estar conectados al mismo tiempo a través de un sensor. Si el ancho de banda consumido por todos los usuarios de un sensor es mayor que el máximo, los tiempos de respuesta de la conexión disminuyen notablemente.

El obtener una correcta distribución de los sensores inalámbricos de red para *iluminar* con Internet una determinada área geográfica se convierte en un problema de optimización de recursos, ya que la distribución a obtener debe realizarse minimizando la función de costos asociados al problema, y maximizando el área de cobertura de los sensores involucrados en la solución. Este problema converge a una categoría de *problema tipo* la cual se conoce como *Facility Location Problem* (FLP) [8]. Este tipo de problema se presenta, por ejemplo, en un conjunto de fábricas (bodegas), que almacenan un bien o servicio, y se encuentran distribuidas dentro de un lugar geográfico, y un conjunto de usuarios cuyo objetivo central es consumir los bienes o servicios almacenados en las fábricas. Además de realizar el consumo de estos bienes o servicios, el consumo debe ser cargado a la fábrica cuya distancia entre su ubicación y la ubicación del usuario sea mínima, lo cual se traduce en que los costos invertidos por el usuario con respecto a consumo y transporte sean también mínimos. La analogía para el problema antes descrito respecto de brindar la conexión a Internet, se resume a obtener una distribución óptima de fábricas (o en este caso sensores) distribuidos en un determinado lugar geográfico. La solución para estos problemas considera tanto la distribución óptima de dichas *fábricas*, como la cantidad mínima necesaria de éstas para lograr el objetivo.

Este artículo se estructura de la siguiente manera. La sección 1 consta de la presente introducción, a continuación, la segunda sección hace una revisión del estado del arte asociado al problema particular; la tercera sección presenta el problema en consideración, mientras que la cuarta sección describe el enfoque utilizado. La quinta sección da cuenta de las pruebas realizadas y los resultados obtenidos; finalmente la última sección presenta las conclusiones que se derivan del trabajo.

2. Estado del arte

El problema asociado a la localización de servicios (FLP) se ha enfrentado desde diversas perspectivas. Lo habitual es hacerlo por medio de programación lineal, como se muestra en [5], donde se menciona la necesidad de resolver el problema a través de una aproximación del esquema primal-dual. Para enfrentar el desafío, modelan el problema usando grafos bipartitos, permitiendo el manejo separado de instalaciones y ciudades, asociando un costo a la apertura de una instalación, y una función que representa el costo de conectar un punto de demanda con una instalación (abierta). La idea es minimizar la función objetivo resultante. El trabajo considera FLP *no capacitado*, para reducir las restricciones y simplificar el problema, lo que repercute en que sea un enfoque menos adecuado cuando se trata con sistemas distribuidos y a gran escala.

Hasan, Jung y Chwa [3] consideran que la apertura de una nueva instalación tiene los mismos costos, independientemente del lugar en donde se produce, y utilizan un algoritmo de programación lineal (LP-rounding). Esta visión es simplificada, y por lo tanto aproximada, pero permite eliminar las restricciones existentes para la ubicación de nuevas instalaciones. Krivitski et al. [8], atacan el problema minimizando una función de costo total; utilizan como entrada una base de datos de puntos, donde cada punto representa a un cliente, un conjunto que contiene posibles ubicaciones, una función de costo, y una configuración que describe las instalaciones en operación. La idea es que dada una función de costo y un conjunto de instalaciones abiertas, la distancia acumulativa de los puntos con las instalaciones cercanas sea minimizada. El esquema es parecido al enfoque anterior, aunque en este caso se agrega una heurística tipo *hill-climbing* para realizar búsquedas locales, obteniendo efectivamente una mejora en los resultados. También existe un trabajo interesante descrito en [4], donde a través de la manipulación de grafos con distancia heredada se van escogiendo subgrafos analizando para cada uno de ellos la distancia de su vecino más cercano, eligiendo la menor para cada subgrafo que modela el problema. En [7], ya más cercana a nuestra propuesta, se estudia la distribución de sensores de red inalámbricos en una determinada área. Ellos concluyen insistiendo en métodos de programación lineal como se describe en [10] o con heurísticas de búsqueda local, como *hill-climbing*. Sin embargo, esta técnica, a pesar de obtener resultados prometedores, dada su naturaleza de búsqueda local, desecha soluciones que pudieran ser eventualmente mejores, limitando negativamente, en este caso, el espacio de soluciones. Chaudhry et al. [1], reducen el problema a la instalación de p fábricas que permitan atender a un número determinado de clientes. El objetivo es minimizar la distancia promedio recorrida por los clientes para llegar a la fábrica más cercana. Este trabajo en esencia resuelve un problema tipo FLP utilizando algoritmos genéticos. Este enfoque permite obtener buenos resultados en tiempos reducidos, aunque en algunos casos se generan soluciones no factibles. Finalmente, el trabajo descrito en [6] a pesar de resolver un problema tipo FLP con algoritmos genéticos, introduce consideraciones al modelo que implican el manejo de restricciones de alto nivel, esto permite obtener buenos resultados,

pero al costo de tener tiempos de procesamiento considerables

El trabajo que aquí se presenta combina algoritmos genéticos y la técnica FLP para proponer soluciones al problema de hacer llegar una señal de Internet a un sector geográfico acotado, relajando la restricción de minimizar el número de sensores, pero manteniendo este aspecto en el cálculo del costo. Adicionalmente, utilizando como base una situación real, se realiza una sintonía de los parámetros del algoritmo para disminuir los costos involucrados. Así, se pretende proponer alternativas que, pese a no cubrir la totalidad de una zona geográfica, ofrecen soluciones con relaciones costo/cobertura razonables, y a su vez ajustables.

3. El problema

La principal dificultad de brindar una señal de Internet es que se necesita contar con una cantidad variable de sensores la cual es directamente proporcional a la cantidad de edificios que existan en cada calle (los edificios actúan como barreras para la señal), además del flujo de usuarios que se pretende abastecer con el servicio. Esto significa que se debe obtener las ubicaciones óptimas para instalar sensores de red, de modo de alcanzar a un conjunto dado de n clientes. Para ello se va a evaluar la generación de un conjunto de posibles ubicaciones en las cuales se puede instalar un sensor, y en donde dicha instalación en la ubicación i incurre en un costo, y cada cliente j debe estar conectado solamente a un sensor, lo que involucra un costo proporcional a la distancia entre el sensor i y el cliente j . El objetivo principal planteado para este problema es obtener la distribución de sensores que minimice la función de costos totales que involucra todos los parámetros antes mencionados.

Algoritmos genéticos

Una técnica frecuentemente utilizada para resolver problemas de optimización son los algoritmos genéticos [2]. La principal ventaja que ofrece esta técnica es que no se necesita contar con información específica del problema a resolver, sino que trabaja con modelos de abstracción que asumen este rol. Por otro lado obtienen de forma simultánea otras soluciones, en vez de calcularlas secuencialmente como las otras técnicas. Y por último, esta técnica está basada en principios evolutivos los cuales son implementados mediante el uso de operadores estocásticos, lo que agrega dinamismo y azar al proceso completo de búsqueda dentro del espacio de soluciones.

Los algoritmos genéticos son métodos adaptativos que pueden usarse para la resolución de problemas de optimización, mediante una analogía con los procesos evolutivos biológicos. La emulación de esos procesos biológicos involucra una codificación de una potencial solución al problema en un *cromosoma*, lo que se traduce en una estructura de datos con características específicas, constituyéndose en una representación, que identifica a los individuos a ser considerados por el algoritmo genético. Luego de definir la representación a utilizar, se debe generar la población inicial, cuya fortaleza radica en la diversidad de

sus elementos (cromosomas) para disminuir el riesgo de los óptimos locales. Una forma habitual de obtener la población inicial es generarla de manera aleatoria. A continuación se define una función de adaptación (*fitness*) que permite evaluar la calidad de una solución, lo que normalmente se mide en términos de costo.

Operadores genéticos: Para este trabajo se consideraron diferentes operadores genéticos. Estos operadores se describen brevemente a continuación:

- **Selección.** La selección se lleva a cabo utilizando el mecanismo de *ruleta* [2]. Esto significa que los individuos con mejores valores de *fitness* tendrán mayor probabilidad de ser escogidos como padres en el proceso de reproducción.
- **Cruzamiento.** El cruzamiento se usa para intercambiar material genético, permitiendo que parte de la información genética de un individuo se combine con parte de la información genética de otro individuo diferente.
- **Mutación.** Al usar este operador genético, se introduce una pequeña variación en la población de manera que se crea nuevo material genético.

4. El enfoque propuesto

En analogía con la técnica FLP descrita anteriormente, cada sensor inalámbrico se convierte en una fábrica, el ancho de banda de cada sensor se convierte en el tamaño de la fábrica y los clientes que debe satisfacer esa fábrica se transforman en la señal entregada por dicho sensor a su vecindad. Respecto de las restricciones para el problema están el que un usuario se conecta a Internet a través de un solo sensor a la vez, los edificios en un sector influyen de forma negativa respecto de la señal, si hay muchos edificios en un sector se requerirá una mayor cantidad de sensores para poder *iluminar* dicho sector con la señal. También, a medida que hay más edificios en un sector, la cantidad de usuarios que requieren Internet aumenta de forma proporcional a la densidad de población.

Para modelar el problema se tuvo en consideración un conjunto de aspectos que se resumen a continuación:

1. Un sensor k posee un determinado ancho de banda, denotado por W_k ; además tiene un radio de alcance A_k que presenta dos valores, uno referido al alcance al interior de las edificaciones, denotado por $A_{int(k)}$ y uno para exteriores, denotado por $A_{ext(k)}$.
2. El área geográfica considerada se representó por medio de un grafo con n nodos. Cada nodo representa la ubicación potencial de un sensor; alternativamente, representa a un grupo de usuarios consumidores de la señal de Internet.
3. El número de sensores considerados es una variable crítica, ya que existe un costo asociado a la instalación de cada nuevo sensor. Se denota s a la cantidad de sensores considerados y δ al costo asociado a cada sensor.
4. Todos los sensores están conectados entre sí.

5. Existen lugares predeterminados para la ubicación de los sensores; en el caso de esta presentación, se eligieron los postes de luz como esos lugares, denotados por P . Una suposición extra es que los postes de luz están separados por una distancia de 20 metros.
6. Cada usuario se conecta a un solo sensor en un instante de tiempo.
7. El alcance de un sensor depende de la cantidad de edificaciones existentes. Si hay edificaciones en torno al sensor, el alcance se calcula utilizando A_{int} . Si no hay edificaciones cercanas al sensor, su alcance será A_{ext} .

Para la resolución del problema, se tomó en cuenta factores como cobertura (denotado por C), distancia (denotado por L) y flujo entre dos nodos, denotado por F . Sea $C_{ij} \in \{0, 1\}$ una función de cobertura, cuyo resultado es 1 si el nodo i es *iluminado* por la señal de un sensor ubicado en el nodo j , y 0 en caso contrario. $C_{kk} = 1$ indica que el nodo k es un sensor. Sea L_{ij} la distancia entre los nodos i y j , y F_{ji} el flujo desde el nodo j al nodo i . Estas consideraciones llevan a las siguientes expresiones:

$Costo = G_0 + \sum_{i=1}^n \sum_{k=1}^n L_{ik} * C_{kk} * \delta$, donde G_0 son los costos iniciales, y δ se asocia a la ubicación de cada sensor. $\sum_{k=1}^n C_{ik} = 1, \forall i \in \{1, \dots, n\}$ refleja la exclusividad en la conexión de un usuario con un único sensor. Lo anterior sujeto a las restricciones $\sum_{k=1}^n C_{kk} = s, \forall k \in S, C_{ik} \leq C_{kk}, \forall i, k \in \{1, \dots, n\}$, y $C_{ik} \in \{0, 1\}, \forall i, k \in \{1, \dots, n\}$. Esto refleja además la cantidad de sensores usados, s ; el hecho de que un sensor es activado a través de la red formada por los sensores activos, evitando la transmisión directa entre nodos que no son sensores. Además se refleja el hecho de que un usuario ubicado en un nodo i puede ser atendido o no por un sensor ubicado en k (valores 1 y 0 respectivamente).

Para el problema particular modelado, F se consideró un valor constante, y se optó por emplear A_{int} como factor de alcance, por tratarse del modelamiento del centro de una ciudad.

5. Pruebas y resultados

El área utilizada para las pruebas comprende el sector céntrico una ciudad arbitraria de tamaño medio. Se consideró una cantidad de diez calles por un lado y ocho por el otro lado, lo que corresponde aproximadamente a 80 hectáreas (ochocientos mil metros cuadrados). La figura 1 esquematiza el área problema.

Cada cuadra (aproximadamente 100 metros) se divide en 6 tramos que pueden ser solicitantes del servicio de Internet. Aquí también están considerados los postes de luz, que son los lugares en donde se ubican físicamente los sensores. De esta manera, una manzana, delimitada por cuatro cuadras, se traduce en 36 puntos eventuales de demanda o servicio. Luego, este sector comprende un total posible de 2880 puntos a ser considerados como posibles demandantes, de los cuales 640 son postes de luz. Utilizando una representación de grafos, cada uno de estos puntos es un nodo, siendo las aristas del grafo la distancia entre los nodos (ver Figura 2). Se genera una matriz triangular con las distancias entre cada par de puntos del modelo, utilizando el algoritmo de Dijkstra del camino



Figura 1. Área problema seleccionada para las pruebas.

mínimo, a la cual se accede durante el proceso posterior de cálculo. El criterio para determinar la bondad de un individuo (solución) está dado por un compromiso entre cobertura y costo de conexión de los s sensores. Se trata en este caso de minimizar la función de costo que toma en cuenta la inversión material, modulada por la cantidad de usuarios que es posible atender. La figura 2 ilustra una distribución de nodos en una manzana arbitraria, que cubre un área de aproximadamente 10000 metros cuadrados.

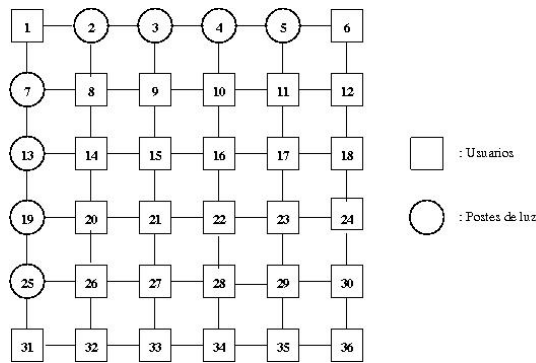


Figura 2. Distribución de nodos en una manzana.

Cada individuo se representa mediante una estructura de datos que se separa en dos partes: la primera, contiene los índices de la ubicación, en el grafo, de cada uno de los s sensores considerados; la segunda parte contiene todos los índices restantes, que corresponden a los nodos que no son sensores.

Lo primero que se hace es calcular aquellos valores que permanecerán constantes durante todo el proceso y que son válidos para todos los individuos de la población; la ya mencionada distancia más corta entre cada par de nodos y la ubicación de los postes de luz.

La población de soluciones considera la restricción de que cada sensor está ubicado en un poste de luz (nodos redondos en la figura 2). Inicialmente se distribuyen los sensores de manera aleatoria, obteniendo los índices desde el conjunto P . El proceso de selección considera el clásico método de la ruleta [2]. Para el cruzamiento, se considera el intercambio de características sólo de la primera parte del cromosoma (que contiene los sensores). Dados dos progenitores escogidos para cruzamiento, si existen sensores repetidos, se reordenan en la cadena, ubicándolos al inicio de la misma, y luego se elige aleatoriamente una subcadena para el intercambio. Esta subcadena no considera la parte inicial de sensores repetidos. Esto permite que los hijos no presenten el problema de asignar un mismo sensor a dos ubicaciones diferentes.

La mutación es implementada cambiando aleatoriamente, con baja probabilidad de ocurrencia, un sensor del cromosoma por otro no considerado en ese individuo (i.e., se desplaza el sensor desde un poste de luz a otro sin sensores). Para conservar los mejores individuos, la estrategia del algoritmo genético considera un 10 % de elitismo. Este 10 % forma parte, junto con los descendientes del cruzamiento, de la siguiente generación de soluciones posibles. Los parámetros utilizados para las pruebas consideraron los valores mostrados en la Tabla 1.

Tabla 1. Parámetros de prueba.

Parámetro	Valor(es)
Número de sensores	50, 80, 90, 100, 150, 200
Operador de selección	Método de la ruleta
Probabilidad de cruzamiento	60 %, 80 %
Probabilidad de mutación	2 %
Porcentaje de elitismo	10 %
Tamaño de la población	20, 80 (individuos)
Número de generaciones	500

La tabla a continuación (Tabla 2), muestra los mejores valores obtenidos para las distintas cantidades de sensores considerados. Las unidades de tiempo y las unidades asociadas al costo son arbitrarias y referenciales; su objetivo es simplemente mostrar los cambios en las magnitudes.

Si bien es cierto que a medida que se aumenta el número de sensores en el modelo, se aumenta también la cantidad de usuarios que es posible atender, esto implica un mayor costo en la habilitación de sensores. A partir de cierto número de sensores, el aumento de los costos tiene una velocidad de crecimiento mayor que el aumento en la cantidad de usuarios atendidos. Esto provoca que el

Tabla 2. Resultados obtenidos.

Prueba	Número sensores	Tiempo	Fitness	Costo	Nodos atendidos
1	50	63.65	3.7383	5200	1391
2	80	41.88	3.6773	6200	1686
3	90	85.74	3.6456	7200	1975
4	100	50.55	3.5101	7280	2074
5	150	71.87	3.7446	8620	2302
6	200	132.69	3.9601	10320	2606

equilibrio no se encuentre en un extremo en cuanto a la cantidad de sensores a considerar, sino que en un punto intermedio, que resulta en este caso ser de 100 sensores. Es esta opción la que consigue una cobertura del 72.01 %, optimizando el fitness de los individuos, de acuerdo a la función definida para este efecto. A modo de ejemplo, en el caso de utilizar 200 sensores, se tiene que la cobertura aumenta en un 18 %, con un incremento en el costo de un 41.75 %. El mejor resultado obtenido se ilustra en la figura 3, donde, a diferencia de la Figura 1 se muestran los puntos donde se ubican los sensores.



Figura 3. Área de prueba ilustrando una posible distribución de sensores.

La figura siguiente (Figura 4) muestra la evolución en la calidad de las soluciones (fitness) para el caso de 100 sensores.

Las diferencias de los resultados alcanzados con 20 y 80 individuos no resultan considerables, aunque los tiempos de ejecución aumentan drásticamente para poblaciones de mayor tamaño. Lo mismo ocurre con la probabilidad de cruce; no influye de manera importante en la calidad de los resultados pero, a medida que aumenta, tiene un efecto negativo en los tiempos de ejecución. Los

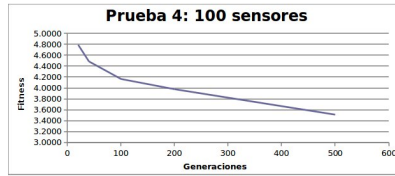


Figura 4. Evolución del fitness para el caso de 100 sensores.

resultados obtenidos comienzan a estabilizarse alrededor de la generación 500, los mejoramientos en la calidad de las soluciones, a partir de ese punto, son poco importantes.

6. Conclusiones

El enfoque de algoritmos genéticos desarrollado en esta propuesta permite analizar el problema de distribución de sensores de red desde una perspectiva diferente. Al utilizar esta estrategia para abordar un problema del tipo Facility Location Problem se pueden considerar aspectos relevantes como son las funciones de oferta y demanda, la cantidad de sensores utilizados y los alcances de estos sensores.

Ante la ausencia de propuestas similares, el modelo que se propone permite priorizar diferentes aspectos involucrados en el diseño de las soluciones; lo que es uno de los aspectos positivos del mecanismo que aquí se sugiere. Estas diferentes priorizaciones, minimizar el costo o maximizar la cobertura, permiten generar cuadros comparativos cuantificados para apoyar una toma de decisiones más informada.

Finalmente, debido a la forma de modelamiento del problema, es posible realizar cambios tanto en los datos iniciales de configuración del área geográfica como en los cálculos de las funciones involucradas. Esto permite reutilizar este modelo para aplicarlo a áreas de diferente tamaño, o a ciudades distintas a la considerada en las pruebas mostradas.

Referencias

1. Chaudhry, S., He, S., and Chaudhry, P.: Solving a class of facility location problems using genetic algorithms. *Expert Systems*, 20(2):86-91 (2003)
2. Floreano, D., Mattiussi, C.: *Bio-Inspired Artificial Intelligence. Theories, Methods, and Technologies*. The MIT Press (2008)
3. Hasan, M., Jung, H., Chwa, K.: Approximation algorithms for connected facility location problems. *Journal of Combinatorial Optimization*, 16(2):155-172 (2008)
4. Hsieh, S.: An efficient parallel strategy for the perfect domination problem on distance-hereditary graphs. *The Journal of Supercomputing*, 39(1):39-57, Springer (2007)

5. Jain, K. and Vazirani, V.: Primal-dual approximation algorithms for metric facility location and k-median problems. *Foundations of Computer Science*, 40th Annual Symposium, pp. 2-13, October (1999)
6. Kratica, J., Stanimirovic, Z., Tasic, D., and Filipovic, V.: Two genetic algorithms for solving the uncapacitated single allocation p-hub median problem. *European Journal of Operational Research*, 182(1):15-28 (2007)
7. Krivitski, D., Schuster, A., and Wolf, R.: A local facility location algorithm for sensor networks. *Lecture Notes in Computer Science*, 3560:368-375, Springer (2005)
8. Krivitski, D., Schuster, A., and Wolf, R.: A local facility location algorithm for large-scale distributed systems. *J. Grid Comput.*, 5(4):361-378, Springer (2007)
9. Negroponte, N.: *Being digital*. Random House Inc., New York, NY, USA (1996)
10. Sviridenko, M.: An improved approximation algorithm for the metric uncapacitated facility location problem, *IPCO*, pp. 240-257, Springer (2002)

Algoritmo alternativo de búsqueda tabú para la solución del problema de diseño económico estadístico de diagramas de control \bar{X} , $\bar{X} - S$

Santiago Omar Caballero Morales y Felipe Trujillo Romero

Universidad Tecnológica de la Mixteca, UTM, Carretera a Acatlima, Km. 2.5,
Huajuapán de León, Oaxaca, México
{scaballero, ftrujillo}@mixteco.utm.mx

Resumen Los diagramas de control son técnicas importantes de control estadístico de la calidad para la pronta detección de fallas en un proceso de manufactura, y la minimización de costos de producción. Diversas técnicas se han usado para encontrar los parámetros de estos diagramas que mejor cumplan con requerimientos económicos y estadísticos. En este artículo se presenta un algoritmo basado en Búsqueda Tabú (BT) para estimar de mejor manera los parámetros para diagramas de control \bar{X} y $\bar{X} - S$. Probando el algoritmo con diferentes modelos de costos y restricciones estadísticas, BT presentó un mejor desempeño cuando se comparó con otras técnicas de estimación como Algoritmos Genéticos (GA) y Métodos Combinatorios (CB).

Palabras clave: Búsqueda tabú, optimización combinatoria, control estadístico de la calidad.

1. Introducción

Los diagramas de control son herramientas usadas para determinar si un proceso de producción se encuentra o no en un estado de control estadístico (y, por lo tanto, si entidades son producidas dentro de los requerimientos de calidad establecidos, los cuales se determinan mediante “límites de control”). Si el atributo de calidad de entidades muestreadas (p.e., peso, longitud, dimensiones, etc.) no se encuentran dentro de límites de control, entonces el proceso se asume que se encuentra en un estado “fuera de control” (entidades defectuosas están siendo producidas). En este caso es necesario encontrar la causa asignable que originó dicho estado (p.e., una falla).

Un diagrama de control necesita de tres parámetros principales: el tamaño de la muestra (n), el intervalo de muestreo (h), y el coeficiente de los límites de control (k). Estos parámetros son ajustados en base a restricciones económicas y estadísticas. Esto porque hay costos y tiempos asociados a la toma de muestras y a la búsqueda de causas asignables: una alta frecuencia de muestreo extendería el tiempo de ciclo del proceso, y dependiendo de la naturaleza de la entidad, a la pérdida de producto. De igual manera, límites de control muy cerrados

incrementaría la frecuencia de detección de fallas y rechazo de entidades que no necesariamente serían de baja calidad.

El Diseño Económico (DE)[2,6] de diagramas de control (la estimación de n, h , y k) considera los costos (en tiempo y dinero) asociados al muestreo y búsqueda/reparación de causas asignables. Sin embargo, el Diseño Económico Estadístico (DEE) [10,11,14,15] adicionalmente considera los requerimientos estadísticos como las probabilidades de error Tipo I (α , detectar un estado “fuera de control” cuando el proceso está bien) y II (β , no detectar el estado “fuera de control” cuando el proceso no está bien).

El diagrama de control \bar{X} es ampliamente usado para controlar la media del atributo de calidad de entidades siendo producidas. En este caso, dicha característica de calidad es una variable numérica. Sin embargo, la variabilidad es un factor importante en el control de un proceso porque materia prima, la habilidad del trabajador, la calibración de máquinas, etc., incrementan la variabilidad del proceso sin afectar de manera significativa la media del mismo [1]. Para monitorear y mantener control sobre ambas, la media y la variabilidad del proceso, el diagrama de control $\bar{X} - R$ es usado, aunque el diagrama R pierde eficiencia para muestras de tamaño $n > 10$ [2]. En este caso, los diagramas de control $\bar{X} - S$ o $\bar{X} - S^2$ son más apropiados [1,11,14].

Encontrar los valores más apropiados para n , h , y k considerando restricciones económicas y estadísticas no es una tarea fácil. Esto dado el número de variables (mínimo de tres, el cual incrementa para el diagrama $\bar{X} - S$) y la complejidad de las funciones de costos. Por lo tanto, el DE y el DEE de estos diagramas se considera un problema de optimización combinatoria.

De entre los métodos o técnicas usados para encontrar soluciones apropiadas para el DE y DEE, se encuentran Hooke y Jeeves (HJ)[3,6,7], Algoritmos Genéticos (Genetic Algorithms, GA) [4,10], y Métodos Combinatorios (CB) [11]. Búsqueda Tabú (BT) [9] es una metaheurística que ha sido utilizada para encontrar mejores soluciones para problemas combinatorios con restricciones económicas (p.e., planeación de horarios y secuencias de producción) [5,8].

En este artículo un algoritmo BT se propone como alternativa para resolver y mejorar el DE y DEE de diagramas de control \bar{X} y $\bar{X} - S$. Tres casos de estudio se consideraron para validar el desempeño del algoritmo, comparándolo con las técnicas de HJ, GA, y CB. Aunque BT obtuvo un desempeño muy similar con HJ, obtuvo mejoras significativas sobre GA y CB para diferentes funciones de costos y restricciones. Los detalles del algoritmo se presentan en la Sección 2, mientras que los resultados obtenidos para cada caso de estudio se presentan en la Sección 3. En la Sección 4 se presentan las conclusiones y el trabajo a futuro.

2. El algoritmo BT

BT es una metaheurística que puede guiar a un algoritmo heurístico de una búsqueda local a un espacio global para encontrar soluciones más allá de la optimalidad local. Puede evitar ciclos en el proceso de búsqueda mediante la restricción de ciertos “movimientos” que pueden hacer al algoritmo re-visitarse.

espacios de soluciones previamente explorados. Estos movimientos se mantienen escondidos o reservados (se mantienen “Tabú”) en una memoria temporal (Lista Tabú) que puede ser actualizada con nuevos movimientos o vaciada (liberada) con diferentes criterios.

El algoritmo BT propuesto (ver Figura 2) considera los siguientes parámetros:

1. Restricciones Estadísticas de α y β (para el diagrama \bar{X} : $\alpha_{\bar{X}} \leq \alpha$ y $\beta_{\bar{X}} \leq \beta$; para el diagrama S , $\alpha_S \leq \alpha$ y $\beta_S \leq \beta$);
2. Los valores asociados a cada costo, tiempo, y atributo, de la función del modelo de costo Z ;
3. T_L , el número de iteraciones del algoritmo BT en las que una solución permanece en la Lista Tabú;
4. $T_1 \leq 20$, el número de diversificaciones (diferentes ejecuciones del algoritmo PRINCIPAL);
5. $T_2 \leq 10$, el número de veces que los parámetros del diagrama de control S no satisface las restricciones de α y β ;
6. $T_3 \leq 10$, el número de iteraciones del algoritmo BT en donde no se han encontrado mejores soluciones.

Para un diagrama \bar{X} sólo $k_{\bar{X}}$ debe ser estimado (y para un diagrama S , sólo k_S). Sin embargo, para el diagrama de control conjunto $\bar{X} - S$, ambas variables deben ser estimadas considerando las restricciones asociadas a cada diagrama individual. Como se muestra en la Figura 2, el algoritmo de solución PRINCIPAL comienza con valores generados aleatoriamente para h , $k_{\bar{X}}$, y k_S , mientras que n es estimada mediante:

$$n = \left(C + \frac{k_{\bar{X}}}{\delta} \right)^2 \quad (1)$$

en donde C es un valor de una distribución normal estandarizada para un valor requerido de β o α [10,13].

El siguiente paso consiste en usar este conjunto inicial de valores para estimar los límites de control Superior (LSC) e Inferior (LIC), α y β para ambos diagramas para verificar si este conjunto representa una solución inicial válida (p.e., cumple con las restricciones estadísticas). Si α o β no cumple con dichas restricciones, el algoritmo procede a re-estimar sólo k_S (por T_2 iteraciones) o el conjunto completo hasta que se obtengan valores que conlleven a cumplir todas las restricciones. Cuando un conjunto válido se obtiene, una **Solución Inicial** es obtenida y evaluada en la función de costo Z , obteniendo un **Costo Inicial**, el cual se convierte en el **Costo Actual** del algoritmo. Con esta solución inicial, el algoritmo BT comienza el el paso de búsqueda, diversificando esta solución en 16 nuevas soluciones que conforman el **Espacio de Búsqueda**. La diversificación es llevada a cabo mediante los movimientos mostrados en la Figura 1, en donde n_0 , h_0 , $k_{\bar{X}_0}$, y k_{S_0} son los elementos de las Solución Inicial. Para el caso de estudio con intervalos de muestreo variables, todos los j intervalos son dependientes de la longitud del primer intervalo h_1 como se define en $h_j = \rho h_{j-1}$ [6], por lo que sólo h_1 es estimado.

Movimientos

$$\begin{aligned}n_1 &= n_0 - \text{random_integer}(0,1) \\n_2 &= n_0 + \text{random_integer}(0,1) \\h_1 &= h_0 - \text{random}(0,0.5) \\h_2 &= h_0 + \text{random}(0,0.5) \\k_1 &= k_{\bar{x}0} - \text{random}(0,0.5) \\k_2 &= k_{\bar{x}0} + \text{random}(0,0.5) \\k_3 &= k_{s0} - \text{random}(0,0.5) \\k_4 &= k_{s0} + \text{random}(0,0.5)\end{aligned}$$

Figura 1. Movimientos de Diversificación.

Para todas las nuevas soluciones, los valores de los límites de control, α 's, y β 's, son estimados para verificar el cumplimiento de las restricciones estadísticas. Si una solución no satisface, entonces $Z = \text{Infinito}$, de lo contrario Z es calculado con los parámetros estimados. Ya que este es un problema de minimización, las soluciones son ordenadas basadas en su valor de costo Z en orden ascendente, por lo que el primer elemento es aquél con el mínimo costo de todas las nuevas soluciones, la **Mejor Solución**.

Para evitar ciclos en la búsqueda (re-visitar soluciones) y encontrar sólo óptimos locales, la **Mejor Solución** es mantenida en una **Lista Tabú (LT)** durante T_L iteraciones del algoritmo BT. Si en la siguiente iteración, la Mejor Solución se encuentra ya en LT, entonces la Mejor Solución sería la segunda (o tercera, o cuarta, etc.) mejor solución en el Espacio de Búsqueda ordenado. Esto permite al algoritmo BT el producir soluciones más diversas incluso si son estimadas a partir de soluciones no tan eficientes.

El **Costo Actual** es actualizado con el de la Mejor Solución, **Costo_MS**, y el algoritmo BT comienza de nuevo a producir 16 nuevas soluciones a partir de la Mejor Solución. Si no se obtiene una Mejor Solución/Costo_MS después de T_3 iteraciones del algoritmo BT, la Mejor Solución es guardada y el algoritmo PRINCIPAL es ejecutado nuevamente (por T_1 iteraciones) para generar nuevos valores iniciales para h , $k_{\bar{x}}$, y k_S . Si T_1 alcanza un límite establecido entonces el algoritmo PRINCIPAL se detiene, terminando el algoritmo BT.

3. Resultados

Primer Caso de Estudio: El algoritmo BT fue probado para resolver el modelo de costo de Rahim [7] para el DE de diagramas de control \bar{X} para el cual se utilizó el método de HJ. Este modelo consideró distribuciones de tiempos de falla *Exponencial* y *Gamma* con intervalos de muestreo fijos. Los datos utilizados para esta prueba fueron: $\mu=180$, $\sigma = \sqrt{10}$, $\delta=5$, $Z_0=0.25$, $Z_1=1$, $D_0=50$, $D_1=950$, $a = 20$, $b = 4.22$, $W = 1100$, $Y = 500$. Este modelo fue adaptado en [10] para considerar distribuciones de tiempo de falla general para el DEE de diagramas \bar{X} con restricciones: $\alpha_{\bar{x}} \leq 0.15$, $\beta_{\bar{x}} \leq 0.20$. Porque en [10] el modelo fue resuelto con GA, el algoritmo BT pudo ser comparado con dos métodos (HJ, GA), obteniendo los resultados mostrados en la Tabla 1. Para este caso, BT tuvo un desempeño muy similar al de HJ y GA, notando que el modelo de costos

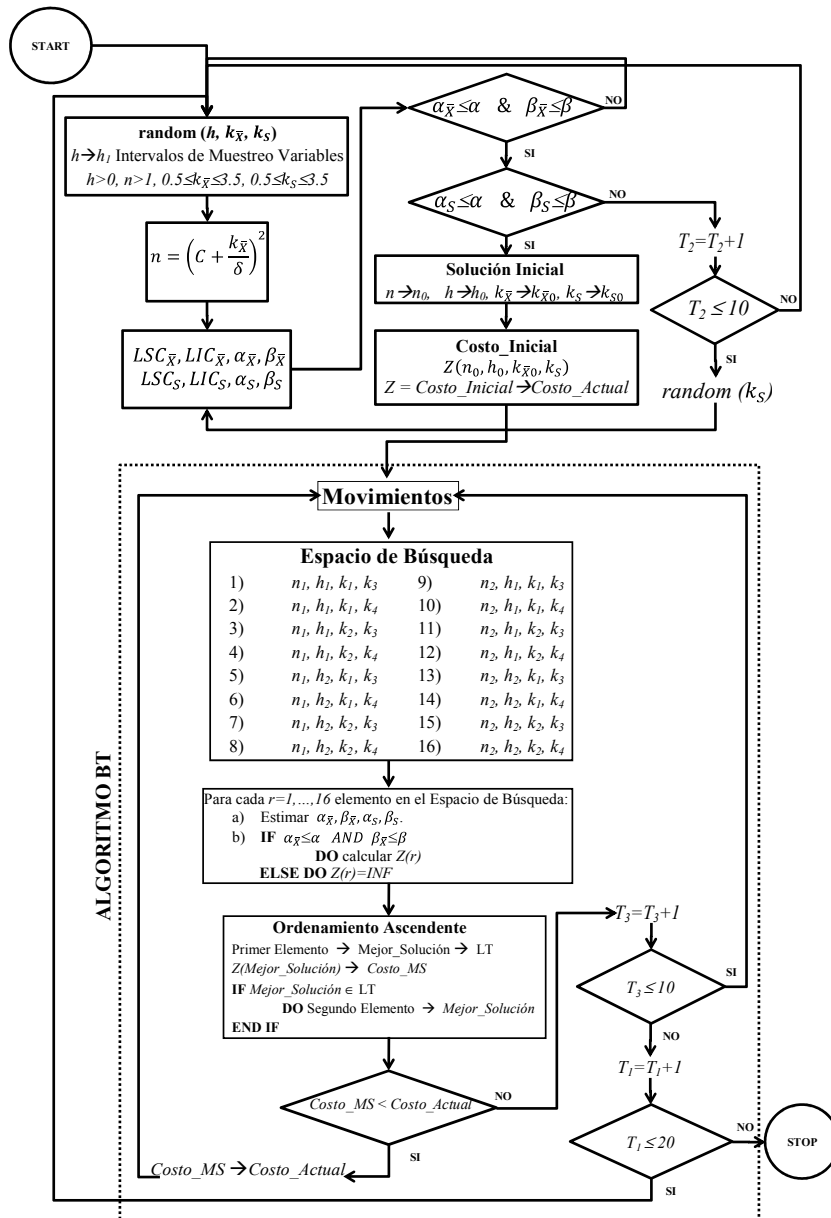


Figura 2. Algoritmo de Solución PRINCIPAL.

no es complejo y que BT tuvo una convergencia a una solución similar (valores de parámetros) similar al de las otras técnicas. Por lo tanto, como alternativa de método de solución, estos resultados dan un nivel de confianza acerca de la habilidad de búsqueda y convergencia de BT.

Otra prueba fue realizada para comparar con GA y distribución *Weibull* para los tiempos de falla e intervalos de muestreo fijos. Los resultados se presentan en

Tabla 1. Desempeño de BT en comparación con HJ y GA: Modelo de Costo de Rahim [7].

Distribución de Fallas	Parámetros	Técnicas	n	h	$k_{\bar{X}}$	$\alpha_{\bar{X}}$	$\beta_{\bar{X}}$	Z
Exponencial	$\lambda=0.2525$	HJ	22	1.380	1.380	0.167	0.168	471.63
		GA	24	1.390	1.440	0.150	0.156	471.96
		BT	23	1.400	1.480	0.140	0.178	472.11
	$\lambda=0.1129$	HJ	24	1.850	1.500	0.134	0.170	332.54
		GA	25	1.910	1.470	0.140	0.153	332.59
		BT	25	1.940	1.500	0.135	0.158	332.60
	$\lambda=0.0505$	HJ	26	2.630	1.570	0.115	0.165	233.28
		GA	27	2.750	1.540	0.125	0.144	233.38
		BT	26	2.550	1.590	0.111	0.170	233.31
Gamma	$Forma=2$ $Escala=0.05$	HJ	28	3.770	1.620	0.106	0.151	174.02
		GA	28	3.870	1.560	0.118	0.139	174.02
		BT	27	3.820	1.550	0.121	0.147	174.10
	$Forma=2$ $Escala=0.08$	HJ	27	3.020	1.580	0.113	0.155	210.43
		GA	28	3.130	1.560	0.119	0.139	210.39
		BT	27	2.940	1.610	0.107	0.162	210.48
	$Forma=2$ $Escala=0.10$	HJ	27	2.740	1.580	0.115	0.153	231.25
		GA	28	2.840	1.560	0.119	0.139	231.17
		BT	27	2.810	1.550	0.121	0.147	231.27

la Tabla 2. Para *Weibull*(2,0,05), con BT una reducción de 1.73% fue lograda para Z , y del 40.3% para $\alpha_{\bar{X}}$ sobre los resultados obtenidos con GA, aunque el valor de $\beta_{\bar{X}}$ se incrementó 26%, $\beta_{\bar{X}} = 0.194$ cumple con $\beta_{\bar{X}} \leq 0.20$. Para *Weibull*(2,0,10) la reducción lograda fue de 2.1% para Z , y del 30% para $\alpha_{\bar{X}}$. Note que la función de costo aumenta su complejidad con la distribución *Weibull*, y que HJ no ha sido abordado para resolver dicho modelo de costo.

Tabla 2. Desempeño de BT en comparación con GA: Modelo de Costo de Rahim [7]-Ruvalcaba [10].

Distribución de Fallas	Parámetros	Técnicas	n	h	$k_{\bar{X}}$	$\alpha_{\bar{X}}$	$\beta_{\bar{X}}$	Z
Weibull	$Forma=2$ $Escala=0.05$	GA	25	1.430	1.480	0.139	0.154	274.90
		BT	27	1.400	1.740	0.083	0.194	270.14
	$Forma=2$ $Escala=0.10$	GA	25	1.130	1.480	0.140	0.133	339.44
		BT	25	1.100	1.650	0.098	0.199	332.29

Segundo Caso de Estudio: Para el DEE de diagramas de control \bar{X} con intervalos de muestreo variables [6,10], BT fue comparado de nuevo con GA con distribuciones *Weibull* y *Gamma*. Los resultados son presentados en la Tabla 3. Para *Weibull*(2,0,05) con $m=7$ intervalos de muestreo, una reducción del 4.69% para Z , 24.74% para el tamaño de muestra, y 57% para $\beta_{\bar{X}}$ fueron logrados. Reducciones similares fueron obtenidas para *Weibull*(2,0,1225). Para *Gamma*(2,0,08) la reducción de costo fue del 0.64% y del 0.27% para *Gamma*(2,0,25). Para $\beta_{\bar{X}}$ las reducciones fueron de 35% y 0.7% respectivamente, y para $\alpha_{\bar{X}}$, 3.7% y 7.7%.

Tercer Caso de Estudio: Para este caso, el DE y DEE de diagramas de control $\bar{X} - S$ fue considerado. Los costos usados fueron aquellos de Davis

Tabla 3. Desempeño de BT en comparación con GA: Modelo de Costo de Rahim [6]-Ruvalcaba [10].

Distribución de Fallas	Parámetros	Técnicas	m	n	h_l	$k_{\bar{x}}$	$\alpha_{\bar{x}}$	$\beta_{\bar{x}}$	Z
Weibull	<i>Forma=2</i>	GA	7	97	2.89	2.45	0.014	0.007	218.03
	<i>Escala=0.05</i>	BT	7	73	2.94	1.50	0.133	0.003	207.81
	<i>Forma=2</i>	GA	6	80	2.07	2.31	0.021	0.015	370.68
	<i>Escala=0.1225</i>	BT	6	62	2.07	1.45	0.148	0.006	359.56
Gamma	<i>Forma=2</i>	GA	4	26	6.94	1.50	0.134	0.146	207.34
	<i>Escala=0.08</i>	BT	4	32	8.01	1.51	0.129	0.095	206.01
	<i>Forma=2</i>	GA	5	27	4.32	1.52	0.129	0.140	352.22
	<i>Escalas=0.25</i>	BT	5	28	4.02	1.56	0.119	0.139	351.27

y Saniga [11] que usaron la función de Lorenzen y Vance [12]. El método de solución fue una técnica combinatoria (CB). Las características de este modelo fueron: distribución *Exponencial* con $\lambda=0.01$, intervalos de muestreo fijos, $\mu=0$, $\sigma=1$, $\mu_1=1.5$, $\sigma_1=2$, $E=0.01$, $T_0=0.1$, $T_1=0.1$, $T_2=0.5$, $C_0=0$, $C_1=100$, $Y=1$, $W=2$, $a=0.5$, $b=0.1$, $d_1=d_2=0$. Los resultados de esta prueba son presentados en la Tabla 4. Para el DEE las restricciones fueron $\alpha < 0.005$ y $\beta < 0.5$.

Tabla 4. Desempeño de BT en comparación con CB: Modelo de Costo de Lorenzen y Vance [12].

Modelo	Técnica	n	h	$k_{\bar{x}}$	k_s	$\alpha_{\bar{x}-s}$	Z
DE	CB	4	1.30	2.00	1.45	0.1386	1.62
	BT	3	1.31	1.24	1.24	0.2150	1.56
DEE	CB	7	1.30	3.00	1.85	0.0049	1.87
	BT	8	1.50	2.83	1.82	0.0046	1.83

Para el DEE una reducción del 2.1% en la función de costo y del 5.1% en $\alpha_{\bar{x}-s}$ fueron obtenidos. Para el caso del DE, en donde ninguna restricción estadística es considerada, una reducción del 3.7% en costo fue obtenida, aunque $\alpha_{\bar{x}-s}$ se incrementó en un 55%. Esto muestra la desventaja del DE cuando se minimizan costos sin considerar las probabilidades de error Tipo I y II.

En la Figura 3 se muestran las gráficas factoriales derivadas de los resultados mostrados en las Tablas 1, 2, 3, y 4. Los factores considerados son:

- Modelo de Costo: cuatro niveles, los cuales corresponden al modelo de Rahim [7] (nivel 1), Rahim [7]-Ruvalcaba [10] (nivel 2), Rahim [6]-Ruvalcaba [10] (nivel 3), y Lorenzen y Vance [12] (nivel 4).
- Distribución: tres niveles, los cuales corresponden a las distribuciones de probabilidad usadas para los tiempos entre falla, que son Exponencial (nivel 1), Gamma (nivel 2), y Weibull (nivel 3).
- Parámetros: tres niveles, los cuales consideran las diferentes configuraciones de parámetros para las distribuciones de probabilidad usadas. Por ejemplo, en la Tabla 1 se utilizaron tres configuraciones para la distribución Exponencial ($\lambda = 0.2525$, 0.1129, y 0.0505). Esto se consideró un factor para evaluar el desempeño del algoritmo para diferentes tasas de fallas.

- Algoritmo: cuatro niveles, identificando cada uno los diferentes algoritmos de solución presentados en los casos de estudio, incluyendo el propuesto. Estos son HJ (nivel 1), GA (nivel 2), BT (nivel 3), y CB (nivel 4).

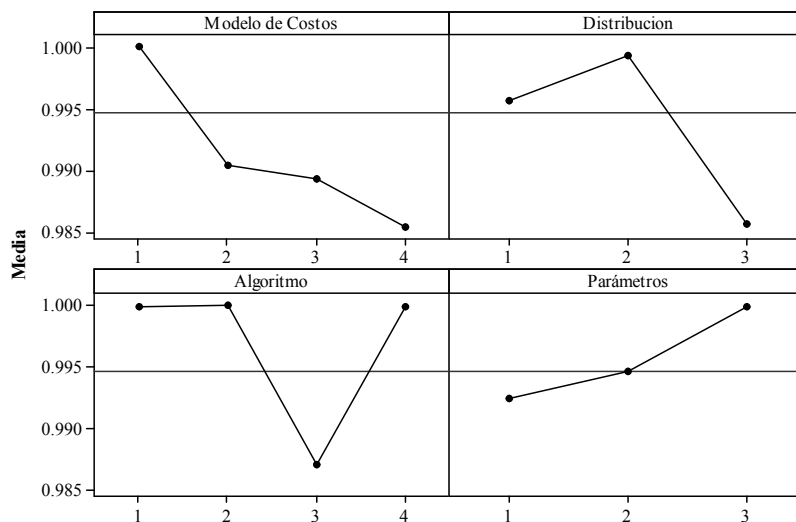


Figura 3. Gráfica de factores principales para Z.

Para la obtención de este análisis los resultados mostrados en las Tablas 1, 2, 3 y 4, se normalizaron a 1.000, tomando como referencia el valor obtenido en la literatura con mayor tiempo. En cuanto a la primer gráfica, correspondiente al **Modelo de Costos**, se observa una tendencia a la baja. Esto es debido a los diferentes costos y restricciones asociados a cada uno, los cuales son dependientes del caso de estudio y no de particular interés para nuestro análisis. Lo mismo sucede con la segunda y tercer gráfica, las cuales corresponden a **Distribución** y **Parámetros**. La cuarta gráfica, **Algoritmo**, es la de principal interés para este trabajo, ya que muestra la reducción promedio de Z considerando los algoritmos utilizados, encontrándose de manera intrínseca la influencia de los factores anteriores. Como se puede observar, es con el algoritmo propuesto de BT (nivel 3) que se obtiene la mayor reducción en Z de manera global.

4. Conclusiones y trabajo futuro

El método de solución propuesto, un algoritmo BT, obtuvo mejoras significativas para el DE y DEE de diagramas de control \bar{X} y $\bar{X}-S$ con diferentes modelos de funciones de costos y distribuciones de probabilidad cuando fue comparado con otros métodos como GA o CB. Aunque para un caso de estudio resuelto con HJ, con intervalos de muestreo fijos y distribución *Exponencial* y *Gamma*, no se obtuvieron beneficios significativos, si se obtuvo una convergencia similar y

por lo tanto se puede considerar BT como un método alternativo de solución. De manera global, como se mostró en la Figura 3, BT si obtuvo beneficios significativos sobre la minimización de las diferentes funciones de costos y distribuciones de probabilidad.

Este trabajo puede ser extendido en los siguientes puntos:

- incorporación de movimientos alternativos para diversificación;
- integración de BT con otro algoritmo (p.e., diseño de un algoritmo híbrido GA-BT);
- realización de pruebas con otros modelos de funciones de costos;
- estudiar la precisión de BT para el DE y DEE de otros diagramas de control (p.e., R , p , np , $EWMA$, etc.);
- consideración de modelos con causas asignables múltiples.

Referencias

1. Collani, V., Sheil, J.: An approach to controlling process variability. *J. Quality Technology* 21, 87–96 (1989)
2. Duncan, A.J.: *Quality Control and Industrial Statistics*. Irwin, Homewood, IL (1986)
3. Hook, R., Jeeves, T.A.: Direct search solution of numerical and statistical problems. *J. ACM* (1961)
4. Kethley, R.B., Peters, M.H.: Extending economic design of p charts to handle user specified constraints using a genetic algorithm. *IIE Transactions* 36, 855–864 (2004)
5. Rahim, A.: A Tabu Search Algorithm for Determining the Economic Design Parameters of an Integrated Production Planning, Quality Control and Preventive Maintenance Policy. In: *Proc. of the 2010 International Conference on Industrial Engineering and Operations Management* (2010)
6. Rahim, M.A.: A generalized model for the economic design of \bar{X} -control charts for production systems with increasing failure rate and early replacement. *Naval Research Logistics* 40, 787–808 (1993)
7. Rahim, M.A., Banerjee, P.K.: The economic design of control chart: A renewal theory approach. *Engineering Optimization* 12, 63–73 (1987)
8. Raza, S., Akgunduz, A., Chen, M.: A tabu search algorithm for solving economic lot scheduling problem. *Journal of Heuristics* 12(6), 413–426 (2006)
9. Reeves, C.R.: *Modern Heuristic Techniques for Combinational Problems*. John Wiley & Sons, Inc. (1993)
10. Ruvalcaba, R.A.: *Economic Design of Quality Control Charts*. MSc. Thesis, University of the Americas Puebla, Mexico (in Spanish) (2004)
11. Saniga, E., Davis, D.J.: Economic-statistical design of \bar{X} and R or \bar{X} and S charts. *J. Quality Technology* 33, 234–241 (2001)
12. Vance, L.C., Lorenzen, T.J.: The economic design of control charts: A unified approach. *Technometrics* 28, 3–11 (1986)
13. Wetherill, G.B., Chiu, W.K.: A simplified scheme for the economic design of \bar{X} charts. *J. Quality Technology* pp. 63–69 (1974)
14. Yang, S.-F.: Economic statistical design of S control charts using taguchi loss function. *Int. J. Quality and Reliability Management* 15, 259–272 (1998)
15. Yu, F.-J., Tsou, C.-S., Huang, K.-I, Wu, Z.: An economic-statistical design of \bar{X} control charts with multiple assignable causes. *J. Quality* 17, 327–338 (2010)

Ontologías, gestión y representación de conocimiento

Introducción a las tecnologías de la Web semántica

Alfredo Villagrán-Olguín, Neil Hernández-Gress y Miguel González Mendoza

Tecnológico de Monterrey
Campus Estado de México, 52926, México
a01165522@itesm.mx, ngress@itesm.mx, mgonza@itesm.mx

Resumen El *World Wide Web Consortium* o Consorcio de la Red Informática Mundial (W3C, por sus siglas en inglés) propone la existencia de una Web Semántica, coexistente con la red mundial actual, que brinde a las computadoras información con significado para ellas. Esta Web Semántica es un conjunto de tecnologías que otorga a las personas la posibilidad de que publiquen información en un formato legible para computadoras, y que provee a éstas la capacidad de procesar semánticamente dicha información. En este trabajo se busca mostrar de forma concisa las tecnologías que constituyen la columna vertebral de la Web Semántica.

Palabras clave: Web semántica, ontología, Linked Data, XML, RDF, SPARQL, OWL.

1. Introduction

Hoy en día, la mayor parte de la información en la Red Informática Mundial o *World Wide Web*¹ está diseñada para ser leída y entendida por personas. Las computadoras carecen de la habilidad de procesar el significado y propósito de la información contenida en los documentos que constituyen la web [4]. Aunque en la actualidad existen poderosas herramientas de búsqueda, éstas aún funcionan sobre un nivel superficial de los documentos: los mecanismos de búsqueda como Bing o Google no se interesan por el significado ni las relaciones de los contenidos entre documentos, sino que buscan palabras semejantes dentro de tales documentos [12].

Para que los sistemas computacionales puedan comunicarse entre sí a nivel semántico, es decir que puedan ejecutar operaciones sofisticadas con el significado de la información, se necesita una web en la que existan documentos cuyo contenido incluya conceptos y relaciones entre estos conceptos. Si se hicieran preguntas específicas sobre un área particular y los documentos en la web contuviesen datos semánticos, podría obtenerse respuestas más precisas que utilizando documentos sin información semántica. Esto es debido a que las tecnologías semánticas son un medio de representación de conocimiento para tal finalidad [6].

¹ En lo sucesivo, llamada simplemente web.

Actualmente existen repositorios que almacenan información semántica de propósito específico [28]. Para utilizar eficientemente esta información, se requiere un lenguaje de consultas y un mecanismo de búsquedas que puedan manipularla. Un mecanismo de búsqueda semántico es una herramienta que se utiliza para encontrar información en una colección de datos semánticos, mientras que un lenguaje de consulta es el medio a través del cual uno especifica la información de interés, contenida en la colección [38].

El presente trabajo tiene el propósito de mostrar las tecnologías fundamentales que constituyen la Web Semántica. Uno comenzará por recordar qué es la representación del conocimiento y cómo se relaciona con la descripción de recursos en la web. Para la descripción de recursos, el W3C propone un lenguaje llamado RDF, que es además un *framework*. Junto con estos recursos, uno puede también describir conjuntos de conceptos y relaciones entre éstos, a través de ontologías. OWL es un conjunto de lenguajes basado en RDF que ayuda al diseño de ontologías. Además, uno puede compartir datos más allá de las restricciones de los sitios en la web y de sus aplicaciones, implementando principios de Linked Data. Linked Data es un conjunto de recomendaciones para utilizar colecciones de datos que contienen información específica acerca de un dominio particular de interés. Finalmente, para consultar colecciones de datos de RDF, OWL y/o Linked Data, uno necesita un lenguaje de consulta y un mecanismo de búsqueda; aquí es donde SPARQL entra en acción y complementa la introducción a las tecnologías de la Web Semántica.

2. Representación del conocimiento

Los agentes inteligentes o autónomos, son sistemas computacionales, por ejemplo, tanto *software*, o *hardware* o ambos, capaces de decidir por ellos mismos qué es lo que necesitan con el fin de cumplir sus objetivos. Los sistemas multi-agentes son ambientes en los que varios agentes interactúan a través de otros, basados en un conjunto de reglas de comunicación e interacción [23].

Debido a que uno espera que los agentes se comuniquen entre sí, hay una necesidad de establecer las bases para que los agentes lleven a cabo dicha tarea. Por lo tanto es necesario representar el conocimiento, al que los agentes tendrán acceso. Entonces, cuando se habla acerca de conocimiento, uno se referirá a la relación entre el conocedor y las proposiciones que serán verdaderas o falsas [8].

2.1. Representación de Esquemas

La representación del conocimiento es una rama de la inteligencia artificial, centrada en dar a los sistemas las facilidades requeridas para el manejo y procesamiento de información reunida en un cierto dominio [17]. Los esquemas más comúnmente usados para representar el conocimiento son clasificados en declarativos y procedimentales. Dentro de los esquemas declarativos uno puede encontrar una subdivisión: redes semánticas y esquemas lógicos.

Esquemas lógicos En este tipo de representación, el conocimiento base, es decir el conocimiento de un dominio particular, es visto como un conjunto de fórmulas lógicas que describen el mundo. Tal representación usa términos como variable, constante, predicado y cuantificadores para explicar datos en un sistema lógico. Con el fin de cambiar la base de conocimiento, la introducción y supresión de fórmulas toma lugar [2].

Redes semánticas El enfoque de los *Frames* o marcos, y el orientado a objetos son otra manera de representar conocimiento. A diferencia del esquema lógico, en el que un grupo de declaraciones lógicas crece y la organización se vuelve crítica, el enfoque semántico considera el agrupamiento de hechos o reglas acerca de objetos al que pertenecen éstos [20]. Tales objetos pueden ir desde cosas concretas (puertas, ventanas, cuartos, etc.) a entidades más abstractas (el tiempo, una fecha, un arribo, una partida). La descripción del mundo toma lugar en términos de objetos (nodos) y asociaciones binarias (aristas etiquetadas). De ahí que el conocimiento base consiste en objetos y sus asociaciones; la modificación del conocimiento base se lleva a cabo a través de la inserción y supresión de objetos; y los cambios se reflejan en las respectivas asociaciones [18].

Esquemas procedimentales Este tipo de esquemas especifican la manera en que una tarea debe ser ejecutada; es decir, el conocimiento procedimental es el conocimiento acerca de habilidades. El patinaje es un buen ejemplo de lo que esta representación puede ser: patinar puede ser una tarea difícil de ejecutar, no obstante puede realizarse.

En Inteligencia Artificial, un sistema experto es un ejemplo de representación procedimental porque consiste en un conjunto de reglas, una memoria, un comparador y un procedimiento que se encarga de resolver los posibles conflictos entre reglas [1].

3. Web semántica

Internet es un conjunto complejo de computadoras interconectadas, a través de la cual viaja información de una máquina a otra. Aunque la web está relacionada a ella, no es un sinónimo de Internet. La web es un conjunto de documentos, imágenes, servicios, etc., que viajan a través de Internet y que se relacionan entre sí a través de enlaces y de Localizadores de Recursos Uniformes (URL, por sus siglas en inglés). Sin embargo, los documentos de la web, transportados en Internet, se encuentran diseñados, en su mayoría, para ser leídos y comprendidos por personas; es decir, las computadoras no son capaces de procesar tales documentos a nivel del significado de éstos. Para ello es que Timothy Berners-Lee ha propuesto el desarrollo de una web semántica: una web en la cual se provea semántica a los documentos procesados por computadoras, y que permita que las máquinas y las personas trabajen de manera cooperativa [4]. La Web Semántica no es una red diferente a la web actual; más bien es un conjunto

subyacente de tecnologías que proporciona a las computadoras la capacidad de describir formalmente objetos en la web, sus propiedades y las relaciones entre ellos. Eso significa que una web semántica se erige como conocimiento representado en la web y las maneras de que sea manipulada por computadoras en vez de por humanos.

3.1. XML

El W3C [41] recomienda un conjunto de especificaciones para trabajar con proyectos de Web Semántica. Estas especificaciones conforman una familia de tecnologías web, entre las cuales XML [43] es la primera. XML es un conjunto de reglas para describir documentos estructurados y programas que procesen tales documentos. A la par, se tienen los esquemas de XML que son utilizados para delimitar la estructura de los documentos en XML[42].

3.2. RDF y esquemas RDF

El *Framework* de Descripción de Recursos o *Resource Description Framework* (RDF, por sus siglas en inglés) y los esquemas RDF [37] son las herramientas básicas para el desarrollo de tecnologías de la web semántica. Mientras que RDF es un lenguaje basado en XML, los esquemas RDF son un conjunto de vocabularios utilizados para describir clases y propiedades, diseñando jerarquías de tales clases y propiedades. Ambos, RDF y los esquemas RDF, son utilizados para describir recursos en la web, creando un modelo de datos como un grafo dirigido. En tales grafos, los recursos son representados por nodos, y las propiedades de los recursos como aristas. Estas aristas son la unión entre los recursos y los valores de sus propiedades [35]. La finalidad de RDF básicamente es mostrar datos a ser manipulados por computadoras, en vez de por humanos, y proporcionar el medio para que las aplicaciones intercambien información.

La idea fundamental de RDF es identificar cualquier tipo de objetos (recursos), ya sean concretos o abstractos, utilizando Identificadores de Recursos Uniformes (URI, por sus siglas en inglés). Una dirección URI es una cadena de caracteres que sirve para referenciar a un particular y único elemento en Internet; es decir, una dirección URI es una dirección *http* válida. De tal manera que un recurso es un objeto identificado con una dirección URI válida, que puede ser descrito a través de sus propiedades.

La propiedades de los recursos son descritos con declaraciones que tienen un formato sujeto-predicado-objeto. El sujeto es el recurso a ser descrito, el predicado es una propiedad o atributo del sujeto, y el objeto es el valor o estado de dicha propiedad. Los sujetos pueden ser recursos o nodos vacíos, los predicados deben ser un recurso, y los objetos pueden ser recursos, nodos vacíos o literales (valores fijos). Un conjunto de declaraciones RDF es llamado un modelo RDF [36]

Una propiedad es un atributo utilizado para describir características acerca de un recurso. Los nodos vacíos representan un recurso y actúan como las variables existencialmente cualificadas en lógica de primer orden. Mientras que las literales son cadenas de caracteres con el valor de una cierta propiedad.

Este formato sujeto-predicado-objeto es conocido como una tripleta RDF. A continuación se muestra un grafo que ejemplifica la descripción de una persona llamada John Smith. Las tripletas conforman un modelo en RDF:

```
<http://www.w3.org/1999/02/22-rdf-syntax-ns#RDF >
  <http://www.w3.org/1999/02/22-rdf-syntax-ns#Description
    about="http://somewhere/JohnSmith/" >
    <http://www.w3.org/2001/vcard-rdf/3.0#Given>John
  </http://www.w3.org/2001/vcard-rdf/3.0#Given>
  <http://www.w3.org/2001/vcard-rdf/3.0#Family>Smith
  </http://www.w3.org/2001/vcard-rdf/3.0#Family>
  </http://www.w3.org/1999/02/22-rdf-syntax-ns#Description>
</http://www.w3.org/1999/02/22-rdf-syntax-ns#RDF>
```

3.3. Vocabularios

Los predicados en un modelo de RDF pueden expresarse gracias a la existencia de vocabularios. Éstos son metadatos; es decir, datos acerca de datos. A través de los vocabularios de RDF uno puede describir propiedades de algún tipo en particular. Los esquemas RDF son una colección de propiedades o términos útiles para describir características acerca de documentos en RDF. Mientras RDF es una recomendación del W3C para crear estructuras de metadatos, los esquemas RDF proporcionan los términos y las propiedades de tales estructuras, y las relaciones entre sus elementos. De ahí que con un esquema de RDF, uno pueda decir que un recurso es de cierta clase o propiedad [37].

A la par de RDF y las clases y propiedades de los esquemas RDF, existen vocabularios o colecciones de metadatos que pueden utilizarse para describir objetos en un dominio particular de discurso. Dos ejemplos de estos vocabularios utilizados en la actualidad son el de la iniciativa Dublin Core (dc) [26] y FOAF [27]. El primero es un conjunto de propiedades útil para describir documentos, y el segundo describe personas y sus relaciones con otras personas, en términos de información personal.

3.4. Formatos de representación de recursos

Como se comentó anteriormente, RDF es un formato de serialización en XML; es decir, un método para convertir estructuras de datos en secuencias de bits manipulables [7]. Sin embargo, existen otros formatos para el diseño de modelos RDF: Notation3, Turtle y N-Triple. Se considera que éstos tienen una estructura de fácil legibilidad para las personas, debido a su sintaxis compacta y cercana al lenguaje natural.

- Notation3 o N3 [29] es un desarrollo directo de Timothy Berners-Lee y la comunidad de la Web Semántica [39]. La finalidad de N3 es proporcionar una serialización de RDF, compacta, no parecida directamente a XML, y de fácil legibilidad para las personas.

- Turtle [40], *Terse RDF Triple Language* o Lenguaje de Tripletas RDF Conciso es un subconjunto de N3, compatible con éste. Turtle permite escribir modelos RDF en una forma compacta, y un documento en Turtle se encuentra conformado por una secuencia de directivas, declaraciones en tripletas y espacios en blanco.
- N-triples [30], a diferencia de N3 y Turtle, es una formato de serialización basado en líneas, y fue diseñado para ser una subconjunto de N3. N-triples representa, por ejemplo, direcciones URI encerradas entre paréntesis angulares, las literales se representan entrecomilladas, y los nodos vacíos como *_:nombreNodo*.

3.5. Ontologías

Una ontología es una colección de términos y conceptos que establecen una área de conocimiento. Uno puede imaginarla como una especificación de un conjunto de objetos, propiedades acerca de estos objetos, y los valores posibles de cada una de éstas. Asimismo, uno puede describir las propiedades entre objetos, gracias a vocabularios específicos [11].

Modelar un dominio de conocimiento, utilizando ontologías, permite que los programas de computadora puedan manipularlas para lograr objetivos particulares. Asimismo, a través de las ontologías, uno puede definir axiomas que determinen la semántica y uso de las entidades representadas en ellas. A nivel formal, las ontologías son declaraciones de una teoría lógica particular.

Antes de XML, uno podía diseñar ontologías con lenguajes como KIF [9], Ontolingua [10] y Frame Logic [14], basados en lógica de primer orden; Loom [16] y PowerLoom [34], basados en la unificación de semántica, la orientación a objetos y los mecanismos de inferencia; OCML [19], similar a Ontolingua y que proporcionaba la producción y deducción de reglas para métodos de resolución de problemas; OKBC [31], basado en la conceptualización de clases; y SHOE [15], basado en la inserción de etiquetas en documentos HTML.

Posterior a XML, han existido lenguajes como XOL [13], que especifica sólo conceptos, taxonomías de conceptos y relaciones binarias, sin un mecanismo de inferencia; DAML-OIL [24], para describir recursos, modelando primitivas comúnmente encontradas en lenguajes basados en *frames*; y OWL [32], del que se tratará a continuación.

OWL es un conjunto de lenguajes basado en XML, Esquemas XML, RDF, Esquemas RDF y DAML-OIL. XML proporciona la sintaxis para la estructura de los documentos, RDF proporciona la semántica para los modelos, y los Esquemas RDF proporcionan vocabularios para las clases y las propiedades. OWL añade propiedades de especificación y clases [33]. Los tres sub-lenguajes que conforman OWL son OWL-Lite, OWL-DL y OWL-FULL.

3.6. SPARQL

SPARQL es el estándar propuesto por el W3C para un lenguaje parecido a SQL, y protocolo de consultas. Por consulta, uno debería entender una manera de obtener información contenida en la Web Semántica; por protocolo, uno

debería entender un medio para comunicar clientes y procesadores de consultas especificadas en el lenguaje SPARQL. Dado que RDF es un formato para describir grafos, SPARQL es un lenguaje útil para mapear éstos.

3.7. Linked data

Tecnologías como RDF, OWL y SPARQL son el fundamento básico de la Web Semántica. Con las dos primeras, las computadoras pueden extraer información de la representación del conocimiento e incluso inferir información no expresada directamente en una ontología. Con SPARQL, se establece un estándar para el diseño de algoritmos e implementaciones para explorar ontologías, de modo que la manipulación de datos por las computadoras no estará restringida a marcos de referencia en particular, sino a un consorcio descentralizado orientado a buscar soluciones comunes y globales para la web: el W3C.

La Web semántica provee un medio para que las computadoras se comuniquen entre sí mediante el intercambio de conocimiento. Específicamente en los últimos cinco años, una nueva tecnología de Web Semántica se ha desarrollado y durante los dos últimos años, las implementaciones de esta tecnología han aparecido en todo el mundo. Esta tecnología es llamada *Linked Data* [28].

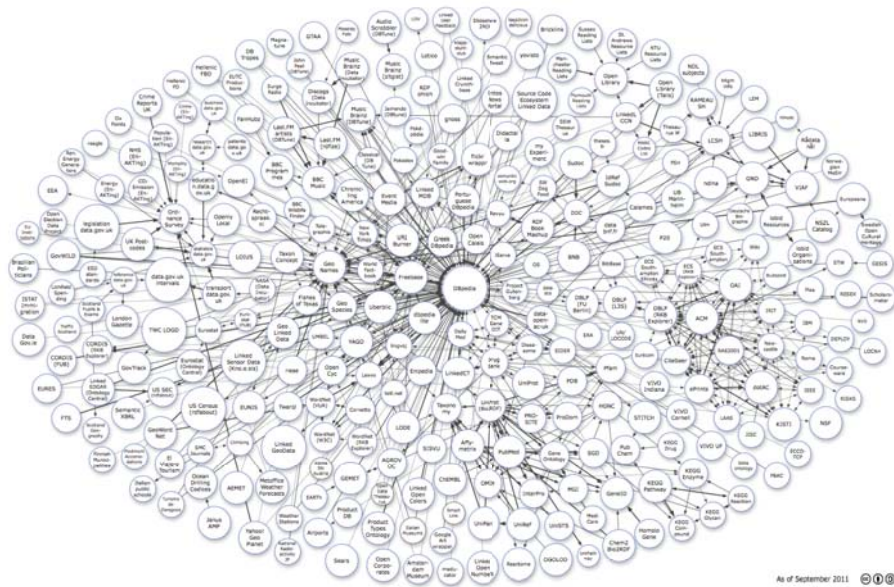


Figura 1. Linked Data graph

El término Linked Data es usado para describir un método de exposición, compartición, y conexión de datos vía direcciones URI referenciables en la web.

Es decir, un proyecto que persigue representar a cada único objeto concreto y abstracto del conocimiento humano como un recurso URI [3], en un formato estándar de descripción de recursos, RDF. El impacto de Linked Data es la creación de las bases de conocimiento en un formato estándar, y estas bases pueden ser compartidas a través de la web. Después de que el conocimiento es compartido, todo lo que queda para determinadas aplicaciones semánticas es explorar estas bases, también con un protocolo estándar (SPARQL).

La figura 1 muestra una imagen que representa los sitios en la web que ya están compartiendo información a través de las recomendaciones de Linked Data, para la publicación de información y desarrollo de la Web Semántica.

En el centro de la imagen se encuentra DBpedia, uno de los sitios principales en la implementación de la tecnología Linked Data. DBpedia es "un esfuerzo de la comunidad [de la web semántica] para extraer información estructurada de Wikipedia y, hacer que esta información este disponible en la web" [25].

4. Conclusiones

La Web Semántica es una red subyacente de la World Wide Web, la cual facilita el camino hacia la Web Semántica, con el fin de lograr el objetivo de que personas y computadoras puedan compartir información sin las preocupaciones de las aplicaciones y los sitios web con sus restricciones.

La Web Semántica es también, un medio por el cual las computadoras pueden intercambiar información a un nivel semántico. Las bases de conocimiento son una especificación de una descripción formal de un área de interés, y las instancias de esta especificación. Con el fin de explotar esta información, se requiere de un lenguaje de consulta y un motor de búsqueda semántica que trabaje con él. Un motor de búsqueda es una herramienta que se usa para buscar información en una base de datos desarrollada semánticamente, mientras que un lenguaje de consulta es el camino por el cual se puede especificar, de la base de datos, en que datos se está interesado.

Tecnologías como XML (conjunto de reglas para describir documentos estructurados), RDF (un marco para describir los recursos en la web), las ontologías (una descripción formal de conceptos y de las relaciones entre ellos), OWL (el conjunto de lenguajes basados en RDF que ayudan a diseñar ontologías), SPARQL (un lenguaje de consulta para lo coincidencia de los grafos escritos en RDF), y Linked Data (un método de exponer, compartir y conectar datos en la Web), son la columna vertebral y el camino hacia el desarrollo de la Web Semántica.

Referencias

1. WRAY, Robert et al.: A Survey of Cognitive and Agent Architectures. University of Michigan, Department of Electrical Engineering and Computer Science. n.d. Web. Jan. 25 (2012)
2. BAADER, Franz, HORROCKS, Ian and SATTLER, Ulrike; Description Logics. In: Handbook of Knowledge Representation, United Kingdom, Elsevier (2008)

3. BERNERS-LEE, Tim: Linked Data. World Wide Web Consortium. August 18th 2009. Web. Jan. 25 (2012)
4. BERNERS-LEE, Tim; HENDLER, James; LASSILA, Ora: The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. *Scientific American*, May (2001)
5. BERNERS-LEE, Tim: example from Tim Berners-Lee talk entitled The next Web of open, linked data at TED2009, February 4 (2009)
6. BERNERS-LEE, Tim et. al.: *Creating a Science of the Web*. Science, August (2006)
7. BOLEY, Harold, and KIFER, Michael: A Guide to the Basic Logic Dialect for Rule Interchange on the Web. In: *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, VOL. 22 (2010)
8. BRACHMAN, Ronald J. and LEVESQUE, Hector J.: *Knowledge Representation and reasoning*. San Francisco CA, Morgan Kauffman Publishers (2004)
9. GENESERETH, M., FIKES, R.: Knowledge interchange format. In: Technical Report Logic-92-1, Computer Science Department, Stanford University, 1992
10. GRUBER, Thomas: A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, 5(2):199-220 (1993)
11. GUARINO, Nicola, and GIARETTA, Pierdaniele: *Ontologies and Knowledge Bases: Towards a Terminological Clarification*. (1995)
12. HAWKING, David: Web search engines. Part 1. *IEEE Computer* 39(6) (2006)
13. KARP, Peter D. and CHAUDHRI, Vinay K. and THOMERE, Jerome F.: XOL: An XML-Based Ontology Exchange Language. Technical Note 559. AI Center, SRI International, 333 Ravenswood Ave., Menlo Park, CA 94025, Jul (1999)
14. KIFER, Michael, LAUSEN, Georg, WU, James: Logical foundations of object-oriented and frame-based languages. *Journal of the ACM* 42 (1995)
15. LUKE, Sean, HEFLIN, Jeff: SHOE 1.01. proposed specification. In: SHOE 1.01, The SHOE Project, April 28, 2000. Web. Jan. 25 (2012)
16. MACGREGOR, R.: Inside the LOOM description classifier. (1991)
17. MYLOPOULOS, John: An overview of Knowledge Representation. In: *Proceedings of the workshop on Data abstraction, databases and conceptual modelling*, New York, ACM (1981)
18. MINSKY, Marvin: *A Framework for Representing Knowledge*. USA, MIT-AI Laboratory Memo 306, June (1974)
19. MOTTA, E.: *Reusable Components for Knowledge Modelling*. Amsterdam, IOS Press (1999)
20. OBRST, Leo; LIU, Howard: *Knowledge Representation, Ontological Engineering, and Topic Maps. XML Topic Maps: Creating and Using Topic Maps for the Web*, Jack Park, Addison-Wesley Professional (2003)
21. PEREZ, Jorge, and ARENAS, Marcelo: Semantics and complexity of SPARQL. *ACM Transactions on Database Systems (TODS) Volume 34* , Issue 3, August (2009)
22. TIJERINO, Yuri A. et al.: A Shape Knowledge Representation Scheme and its Application on a Multi-modal Interface for a Virtual Space Teleconferencing System. *Robot and Human Communication, Proceedings., 4th IEEE International Workshop on DOI, TOKYO* (1995)
23. WEISS, Gerhard: *Multiagent Systems, A Modern Approach to Distributed Modern Approach to Artificial Intelligence*. Cambridge-London: MIT Press (1999)
24. The DARPA Agent Markup Language. DARPA's Information Exploitation Office, n.d. Web. 25 Jan (2012)

25. DBpedia. Freie Universität Berlin, Openlink Software and Universität Leipzig, 2011-11-08. Web. 25 Jan (2012)
26. dublicore. The Dublin Core Metadata Initiative, n.d. Web. Jan. 25 (2012)
27. FOAF. The Friend of a Friend project, n.d. Web. Jan. 25 (2012)
28. Linked Data. The Linking Open Data Project, n.d. Web. Jan. 25 (2012)
29. Notation3. The World Wide Web Consortium, September 2, 2011. Web. Jan. 25 (2012)
30. NTriples. The World Wide Web Consortium, July 5, 2001. Web. Jan. 25 (2012)
31. OKBC. OKBC Working Group, n.d. Web. Jan. 25 (2012)
32. OWL. The World Wide Web Consortium, October 15, 2007. Web. Jan. 25 (2012)
33. OWL Features. The World Wide Web Consortium, November 12, 2009. Web. Jan. 25 (2012)
34. PowerLoom. The Artificial Intelligence research group at the University of Southern California's Information Sciences Institute. July 18, 2007. Web. Jan. 25 (2012)
35. RDF. The World Wide Web Consortium, February 10, 2004. Web. Jan. 25 (2012)
36. RDF Primer. The World Wide Web Consortium, February 10, 2004. Web. Jan. 25 (2012)
37. RDF-Schema. The World Wide Web Consortium, February 10, 2004. Web. Jan. 25 (2012)
38. SPARQL. The World Wide Web Consortium, January 15, 2008. Web. Jan. 25 (2012)
39. Semantic Web. The World Wide Web Consortium, November 7, 2011. Web. Jan. 25 (2012)
40. Turtle. The World Wide Web Consortium, March 28, 2011. Web. Jan. 25 (2012)
41. W3C. The World Wide Web Consortium, January 25, 2012. Web. Jan. 25 (2012)
42. XML Schema. The World Wide Web Consortium, October 28, 2004. Web. Jan. 25 (2012)
43. XML. The World Wide Web Consortium, November 26, 2008. Web. Jan. 25 (2012)

Geontomex: una ontología espacial de México para la desambiguación de topónimos

Belém Priego Sánchez, María J. Somodevilla García, Ivo H. Pineda Torres
y Jaime Hernández Gómez

Benemérita Universidad Autónoma de Puebla,
Av. San Claudio 14 Sur, Col. San Manuel Puebla, México
{belemps, mariajsomodevilla, ivopinedatorres, kngjaime }@gmail.com

Resumen En este artículo se presenta una ontología espacial de la República Mexicana como repositorio de sentidos para la tarea de desambiguación de topónimos. La ontología incluye la representación de objetos geográficos naturales y artificiales. El uso de ontologías ayuda a solventar problemas tales como encontrar el verdadero sentido de las palabras, incluyendo en un solo repositorio todos los conceptos y relaciones del dominio de trabajo. De esta manera se evitan errores en el manejo de la información y es posible unificar el lenguaje de la comunicación en función de sus diferentes sentidos semánticos para desambiguar. La ontología espacial Geontomex fue desarrollada en el gestor de ontologías Protégé y posteriormente validada con el razonador RacerPro para garantizar la consistencia de la misma.

Palabras clave: Ontología, recuperación de información geográfica, relaciones espaciales, topónimo, desambiguación.

1. Introducción

La desambiguación de topónimos (DT) es una de las tareas de la recuperación de información (Information Retrieval, IR), más específicamente en la recuperación de información geográfica (Geographical Information Retrieval, GIR) y búsqueda de respuestas (Question Answering, QA), incluyendo además, la generación de mapas. DT tiene como objetivo relacionar nombres de lugares con su representación geográfica. La Geo-información es abundante en la Web y bibliotecas digitales, por ejemplo: colecciones de fotografías geo-referenciadas (Flickr), noticias, y bases de datos de información demográfica (En México a cargo del Instituto Nacional de Estadística, Geografía e Informática, INEGI). Aproximadamente el 80% de las páginas web contienen referencias a lugares [1], mucha de la información necesaria está relacionada a un contexto geográfico dado, por ejemplo: encontrar los restaurantes más *cercanos*, encontrar noticias acerca de *México*, así como encontrar fotografías tomadas en *Cholula*, entre otras. Se ha reportado también en la bibliografía, que aproximadamente el 20% de las consultas en la Web son de naturaleza local [2]; esto quiere decir que la información geográfica está omnipresente.

GIR pertenece a una rama de la recuperación de información, que incluye todas las tareas de investigación que tradicionalmente forman el núcleo de la IR, pero además con un énfasis en la información geográfica y espacial. En la tarea de GIR, la mayoría de las peticiones de los usuarios son del tipo "X" en "P" donde P representa un nombre de lugar y X, la parte temática de la consulta [14]. GIR aborda dificultades de IR, tales como: ambigüedad Geográfica (topónimos) por ejemplo: existe una *catedral* en *St. Paul* en *Londres* y otra en *Sao Paulo*, o el caso de regiones geográficas mal definidas: "cerca del este", regiones geográficas complejas: "cerca de ciudades rusas" o "a lo largo de la costa mediterránea", aspectos multilingües como son: "GreaterLisbon" en inglés significa lo mismo que "Grande Lisboa" en portugués o que "GroBraunLissabon" en alemán y la existencia de la granularidad en las referencias a países: "al norte de Italia".

La información puede ser geo-referenciada por nombres de lugares o por coordenadas geográficas. Para este caso tenemos las siguientes clasificaciones: formal e informal y mediante un recurso (repositorio de sentidos). El repositorio de sentidos hace referencia a la utilización de diccionarios, tesauros u ontologías, en el cual se indiquen los distintos sentidos de las palabras [14].

Existen diferentes ontologías de topónimos reportadas en la bibliografía tales como Geo-WordNet [11], GeoNames [12], TRIPOD [13], por citar algunas, pero la mayoría de estas incluyen topónimos del lenguaje inglés. También se han comenzado a reportar algunas ontologías de topónimos para los lenguajes alemán [15], portugués [16], chino [17], entre otras. Sin embargo, no se han encontrado reportados en la bibliografía trabajos relacionados con ontologías de topónimos en español. Por lo tanto se hace imperativo el desarrollo de ontologías tales como la que proponemos, para soportar la tarea de resolución de topónimos presentes en los documentos escritos en español que se recuperan por peticiones en la Web.

En este trabajo se propone una ontología espacial de México para la tarea de desambiguación de topónimos. Una ontología espacial es una estructura que permite describir adecuadamente las características específicas del espacio geográfico. Una ontología proporciona un vocabulario de clases y relaciones para describir un ámbito determinado; éstas permiten que las máquinas puedan intercambiar información de forma efectiva y eficiente. Para ello proporcionan formalismos y estructuran la información permitiendo un cierto grado de razonamiento automático.

El resto del artículo está estructurado en las siguientes secciones; en la Sección 2 se explica la tarea de desambiguación de topónimos. El repositorio de sentidos se discute en la Sección 3, con el propósito de sentar las bases para la descripción de la propuesta presentada en la Sección 4. Finalmente, en la Sección 5 se reportan las conclusiones y se presenta el posible trabajo a futuro.

2. Desambiguación de topónimos

Los topónimos pueden ser ambiguos y además tener algún tipo de ambigüedad ya sea GEO/GEO o GEO/NO-GEO. En el caso de la ambigüedad de topónimos GEO/GEO un topónimo representa varios lugares geográficos, por ejemplo, "Tripoli" que es el nombre de 16 lugares en el mundo. En el caso de la ambigüedad

de topónimos GEO/NO-GEO, un topónimo puede referirse a entidades geográficas y no geográficas, por ejemplo, “Benito Juárez” representa a una persona y también a lugares; otro ejemplo, “Java” es una isla indonesia y a su vez un lenguaje de programación. Para ambos tipos de ambigüedad, los dominios de aplicación son la extracción y recuperación de información.

La desambiguación de topónimos constituye una de las tareas importantes dentro de la recuperación de información geográfica. Recientemente, ha habido un gran interés en el problema de desambiguación de topónimos desde distintas perspectivas como el desarrollo de recursos para la evaluación de los métodos de desambiguación de topónimos [3] y el uso de estos métodos para mejorar la resolución del alcance (*scope*) geográfico en documentos electrónicos [4], entre los trabajos más representativos. No sería posible estudiar la ambigüedad de los topónimos sin estudiar también los recursos que se involucran en el proceso, como bases de datos, diccionarios y otros que se usan para encontrar los significados diferentes de una palabra. Considerando que los métodos para la desambiguación de topónimos son de diferente naturaleza, estos aún tienen factores en común. La mayoría están influenciados por dos fases principales, la primera fase es extraer los referentes candidatos y la segunda fase es seleccionar el referente correcto.

Dada la importancia del tema desarrollado y una vez realizada la revisión bibliográfica se encontraron muy pocos trabajos sobre desambiguación de topónimos para el idioma español y eso precisamente fue lo que motivó el desarrollo del presente trabajo. En la siguiente sección se describe el repositorio de sentidos: la ontología espacial, que describe formalmente los distintos sentidos de los objetos geográficos naturales y artificiales en base a los conceptos y relaciones del dominio.

3. Ontologías

Las ontologías permiten que las máquinas puedan intercambiar información de forma efectiva y eficiente. Para ello proporcionan formalismos y estructuran la información permitiendo un cierto grado de razonamiento automático. Gruber [5] creó una de las definiciones más citadas del concepto de ontología en el ámbito de la informática: “una especificación explícita y formal sobre una conceptualización compartida”.

Gruber [6] es también uno de los autores más citados al identificar los cinco componentes básicos del modelado de ontologías, los cuales se enuncian a continuación:

- Conceptos. Son las ideas básicas que se intentan formalizar.
- Relaciones. Representan las interacciones y los enlaces entre los conceptos del dominio.
- Funciones. Son casos especiales de relaciones donde se identifican elementos mediante el cálculo de una función que considera varios elementos de la ontología.
- Instancias. Se usan para representar elementos determinados en una ontología.

- Axiomas. Los axiomas formales sirven para modelar sentencias que son siempre ciertas. Normalmente, se usan para representar conocimiento que no puede ser formalmente definido por los componentes descritos anteriormente consiguiendo así una mayor capacidad expresiva del dominio. Además, también se usan para verificar la consistencia de la propia ontología.

3.1. Ontologías espaciales

Las Ontologías Espaciales, de acuerdo a [7], son una extensión de la Lógica Descriptiva (DL), con un dominio concreto para la dimensión espacial (es decir, considera objetos espaciales tales como puntos, líneas, polígonos), para así permitir la combinación de representación del conocimiento y el razonamiento espacial dentro de un paradigma único. El dominio concreto está definido por un conjunto de predicados representando relaciones topológicas entre objetos. La habilidad de definir roles topológicos facilita la especificación de conceptos y objetos espaciales. También provee acceso algoritmos de razonamiento espacial que permiten la extensión del razonamiento terminológico de la dimensión espacial.

Algunos investigadores, como Spaccapietra [8], dividen esta Ontología en espacio y tiempo; las ontologías de tiempo definen los conceptos que son usados en un tiempo especificado y elementos temporales, como son las instancias, intervalos, cronómetros, entre otros, y relaciones temporales como son precedentes, antecedentes, entre los más significativos, pero de igual manera son denominados espaciales. Debido a la aparición de una gran cantidad de información geográfica y mapas en Internet, de casi todos los sitios posibles sobre la tierra, aparecen los Servicios de la Web Semántica [9] que convienen a un tipo de tecnologías más elaboradas, en un mundo donde se cree que más del 80% de los datos tiene un componente geográfico, como lo son las nuevas aplicaciones de mapas publicadas en el Web. Los mapas web muestran recientemente grandes crecimientos, su integración dentro del dominio espacial aparece como un paso esencial hacia la adopción de la tecnología SWS (Shore Wireless Service). Sin embargo, el espacio geográfico como un único pero total dominio encuadrado tiene especificaciones que describen semánticas más reconocidas. Además, los Sistemas de Información Geográfica (GIS) necesitan adoptar habilidades humanas cognitivas de representación espacial y razonamiento.

La aparición de las Ontologías Espaciales sirve como soporte a este tipo de tecnologías para acceder y compartir información utilizando como componente esencial a los aspectos espaciales y temporales.

4. Geontomex: desarrollo y descripción

El desarrollo de la ontología geográfica propuesta en este artículo, se enmarca en un proyecto global ambicioso, cuyo objetivo es de ser tomado como repositorio de sentidos para resolver la tarea de desambiguación topónimos. Cabe mencionar que esta ontología puede ser utilizada para desambiguar topónimos en consultas a la Web, relacionadas por ejemplo con: el ejercicio y la promoción del turismo en México,

aplicaciones de cambio climático, realización de planeaciones urbanas y desarrollo de planes estratégicos de tipo económico-sociales entre otras. Dentro de este proyecto, uno de los aspectos a desarrollar es la implementación de una ontología que permita identificar el topónimo correcto en un contexto (*corpus*) y a su vez mediante la ontología obtener su posición geográfica (latitud y longitud).

Para cumplir con esta meta, este artículo presenta una ontología espacial que describe el espacio geográfico considerando los objetos geográficos naturales y artificiales, entendiendo por objetos naturales aquellos que fueron creados por la naturaleza (bahía, golfo, lago, etc.) y por objetos artificiales los que son creados por el hombre (puente, aeropuerto, ferrocarril, etc.). Hasta este momento la ontología propuesta sólo comprende objetos geográficos de la República Mexicana, debido a que en otros países de habla hispana la división política varía de acuerdo a cada país y no se garantizaba la consistencia de la ontología durante el proceso de validación con el razonador lógico RacerPro.

Existen varios lenguajes ontológicos para implementar ontologías, los cuales proporcionan distintos niveles de formalismo y facilidad de razonamiento. El lenguaje OWL¹ (Ontology Web Language), estandarizado por el W3C (World Wide Web Consortium), permite definir ontologías con varios niveles de detalle. Dicho lenguaje se puede categorizar en tres especies o sublenguajes: OWL-Lite, OWL-DL y OWL-Full. La ontología espacial se implementó en Protégé² empleando el sublenguaje OWL-DL, debido a que está diseñado para aquellos usuarios que requieren máxima expresividad conservando completitud computacional (se garantiza que todas las conclusiones sean computables) y resolubilidad (todos los cálculos se resolverán en un tiempo finito) [10]. Una de las ventajas de utilizar este Protégé es que cuenta con el manejo de instancias sobre las clases, así como restricciones para generar éstas.

4.1. Desarrollo: Geontomex

En el desarrollo de ontologías, el primer paso es identificar la información que se quiere representar. Lo más adecuado es tomar como base de conocimiento de expertos en el dominio en cuestión, aprovechando posibles categorizaciones o clasificaciones ya existentes.

Para la definición de aspectos genéricos de la ontología, han servido como base la ontología presentada en [10], una ontología mixta que proporciona un vocabulario de clases y relaciones para describir un área específica. En este caso el espacio geográfico incluye un análisis de la distribución de 500 millones de hispanohablantes estimados en el mundo.

Conceptos (Jerarquía de Clases)

La ontología se desarrolla a partir de la jerarquía de clases que se muestran en la figura 1. En esta figura se pueden distinguir tres clases de alto nivel: *Extension_Geografica*, *Estructura_Geografica* y *Localizacion*.

Las clases constituyen las unidades básicas de la ontología que se pretende formalizar, a continuación se describen las clases de alto nivel de Geontomex:

¹ www.w3.org/TR/owl-features

² protege.stanford.edu

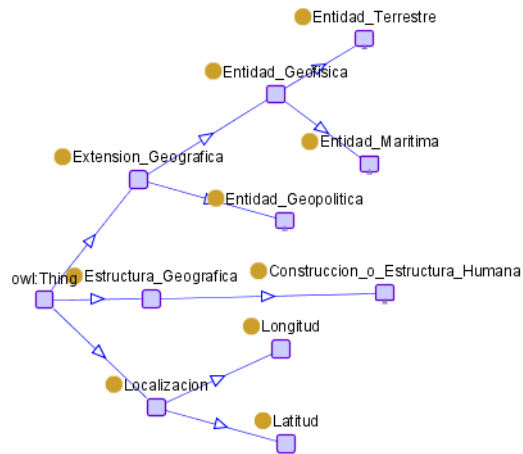


Figura 1. Taxonomía de la Ontología Espacial: Geontomex

- *Estructura_Geografica*: representa la clasificación más habitual de áreas o espacios artificiales creados por el ser humano. Cuenta con una subclase: *Construcion_o_Estructura_Humana* que a su vez tiene diez subclases: *Aeropuerto*, *Calle_o_Carretera*, *Camino*, *Canal*, *Ferrocarril*, *Monumento*, *Parque_Natural*, *Presa*, *Puente* y *Puerto*.
- *Extension_Geografica*: representa los espacios geográficos naturales; dentro de esta clase se incluyen las subclases: *Entidad_Geofisica*, que define los espacios naturales marítimos y terrestres de México y la *Entidad_Geopolitica*, que representa la división política de México.
- *Localizacion*: representa una manera de definir la localización de un lugar.

Relaciones

• Propiedades de objeto

A partir de la jerarquía de clases presentada en la figura 1, se definen una serie de propiedades de objeto para, principalmente relacionar todas las clases de la ontología con la entidad *Localizacion*. A partir de las relaciones, es posible definir todos los aspectos que se quieren relacionar con la localización de los topónimos. A continuación en la Tabla 1 se detallan las relaciones descritas como propiedades de objeto incluidas en la ontología.

• Propiedades de datos

Además de las propiedades que relacionan las diferentes entidades de la ontología, es necesario crear propiedades de datos que describen dichas entidades. Las principales en esta ontología son las que describen la entidad *Localizacion* y son las siguientes *latitud* y *longitud*. Estas son consideradas propiedades funcionales ya que describen los valores de la localización de un topónimo.

Tabla 1. Propiedades de objeto de Geontomex.

Propiedad de objeto	Tipo	Descripción
<i>contieneLocalizacion</i>	Transitiva A,B,C : localizaciones Si $A \subset B \wedge B \subset C$, $\Rightarrow A \subset C$	Relaciona dos localizaciones para indicar que una localización puede contener a otra de longitud más reducida. Propiedad inversa de <i>formaParteDeLocalización</i> .
<i>formaParteDeLocalizacion</i>	Transitiva A,B,C : localizaciones Si $A \subset B \wedge B \subset C$, $\Rightarrow A \subset C$	Relaciona dos localizaciones para indicar que puede formar parte de otra de mayor longitud. Propiedad inversa de <i>contieneLocalización</i> .
<i>tienePuntoInicio</i>	Funcional l: localización A: punto inicial $l = l(A)$	Relaciona una localización con su latitud y longitud, inicial. Subpropiedad de <i>formaParteDeLocalización</i> .
<i>tienePuntoFinal</i>	Funcional l: localización B: punto final $l = l(A)$	Relaciona una localización con su latitud y longitud del punto de finalización. Subpropiedad de <i>formaParteDeLocalización</i> .
<i>estaADistancia</i>	Simétrica A,B : puntos d: distancia $d(A,B) = d(B,A)$	Relaciona dos localizaciones para el cálculo de la distancia existente entre ellas.

Instancias

Para comprobar la utilidad de la ontología, se incluyeron instancias que verifican el correcto funcionamiento de Geontomex. Geontomex está compuesta por 2483 instancias, donde cada una comprende a un topónimo diferente.

Para la tarea de desambiguación de topónimos se está utilizando como contexto un Corpus de noticias multilingüe (incluye los idiomas: español, inglés, francés, italiano y portugués) de la tarea de búsqueda de respuestas (QA) de la iniciativa CLEF³, con una colección total de 4264 documentos de los años 2003 a 2005. Para la tarea que se

³ <http://www.clef-initiative.eu/>

está realizando sólo se ocupó el corpus en el idioma español, el cual consta de 731 documentos. El corpus anterior comprende 216102 noticias de todo el mundo y cada una contiene topónimos.

Geontomex cubre 38% de noticias del total mencionado anteriormente, además de un 24.12% de topónimos diferentes contenidos en esas noticias. Esto es con respecto a las noticias de todo el mundo, así que se prosiguió a separar las noticias de México teniendo ahora un subconjunto del corpus y un total de 6633 noticias con contenidos relacionados con México. Geontomex abarca entonces un 99% de las noticias de México representando un 18.85% de topónimos diferentes que aparecen en esas noticias. Con estos resultados, nos damos cuenta que se cubre la mayor parte del subconjunto del corpus de noticias de México y de esta forma se verifica que Geontomex tiene un buen funcionamiento y considera a la mayor parte de los topónimos.

Axiomas

Los axiomas están en desarrollo, teniendo como principal axioma de Geontomex la relación espacial “cerca”, esta relación se está diseñando y se tiene previsto incorporarla para seguir verificando la consistencia de la ontología. Este axioma trata el concepto de ambigüedad espacial y se está programando con las relaciones básicas de la ontología que son: *is_part_of* e *is_a*, teniendo como base la siguiente expresión:

$$\text{Si } "A" \wedge "B" \text{ is_part_of } "C" \Rightarrow "A \wedge B" \text{ están cerca.}$$

Donde: A, B, C son topónimos de la Ontología produce el siguiente axioma:

$$\text{cerca}(A,B) := \text{cross}(A,B) \vee \text{inside}(\{A,B\}, C) \vee \text{touch}(A,B)$$

4.2. Validación de Geontomex

Para garantizar la consistencia de Geontomex, ésta se validó utilizando un razonador espacial. En este proceso de validación, la información acerca de los objetos en el espacio y sus interrelaciones son recogidas por varios medios, tales como medidas, observaciones, o inferencia, y se utilizan para llegar a conclusiones válidas conforme a las relaciones de objeto o para determinar la forma de realizar una tarea. El razonamiento espacial es usado para inferir todas las relaciones posibles entre un conjunto de objetos usando un subconjunto de las relaciones especificadas. RacerPro⁴ es un razonador utilizado tanto para Lógica Descriptiva Básica, como para muy expresiva y espacial, por este motivo Geontomex fue validada mediante este razonador. Además, también puede ser usado como un sistema para gestionar las ontologías de la Web Semántica basadas sobre OWL, es decir, puede ser usado como un motor para editores ontológicos como Protégé.

⁴ Sitio oficial RacerPro: <http://www.racer-systems.com/>

El editor de RacerPro utilizado para la validación, fue RacerPorter⁵ en el cual se pueden cargar bases de conocimiento, conmutar entre diferentes taxonomías, inspeccionar las instancias, visualizar TBoxes y ABoxes, manipular los servicios, entre otros servicios que la interfaz maneja, para la validación, razonamiento espacial y verificación de consistencia de ontologías.

5. Conclusión y trabajo futuro

En este artículo se presentó Geontomex, una ontología espacial que sirve como repositorio de sentidos para una de las tareas importantes de la recuperación de información geográfica como lo es la desambiguación de topónimos para el idioma español en particular para la República Mexicana; cabe resaltar que existen pocas investigaciones sobre esta temática para el idioma español en comparación con otros idiomas, en específico con el idioma inglés.

Se está trabajando en el enriquecimiento del corpus para ser utilizado como contexto a través de la utilización de técnicas de bootstrapping. El corpus enriquecido se utilizará entonces como conjunto de prueba para generar un modelo de clasificación de topónimos tipo GEO/GEO Y GEO/NOGEO, el cual se plantea como meta final de este proyecto de investigación.

Referencias

1. Spink, A., Wolfram, D., Jansen, M.B.J., Saracevic, T.: Searching the web: the public and their queries. *Journal of the American Society for Information Science and Technology* 52(3), 226–234 (2001)
2. Sanderson, M., Kohler, J.: Analyzing geographic queries. In: *Workshop on Geographic Information Retrieval SIGIR* (2004)
3. Leidner, J.L.: *Toponym Resolution in Text: Annotation, Evaluation and Applications of Spatial Grounding of Place Names*. Universal Press, Boca Raton, FL, USA (2008)
4. Andogah, G.: *Geographically Constrained Information Retrieval*. PhD thesis, University of Groningen, Groningen, Netherlands (2010)
5. Gruber T.R.: Towards Principles for the Design of Ontologies Used for Knowledge Sharing. In *Formal Ontology in Conceptual Analysis and Knowledge Representation*, Deventer, The Netherlands, (1993)
6. Gruber T.R.: A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, 5(2):199–220 (1993)
7. Haarslev V., Lutz C., Moller R.: Foundations of Spatioterminological Reasoning with Description Logics. *Proceedings of the sixth Int. Conf. On Principles of Knowledge Representation and Reasoning (KR'98)*, A.G. Cohn et al, pp. 112-123 (1998)
8. Spaccapietra S., Cullot N., Parent C., Vangenot C.: On Spatial Ontologies. In: *6th Brazilian Symposium on GeoInformatics, GeoInfo, Campos do Jordao, Brazil, Noviembre 22-24* (2004)
9. Tanasescu V., Gugliotta A., Domingue J., Davies R., Gutiérrez-Villarías L., Rowlett M., Richardson M., Stincic S.: *A Semantic Web Services GIS based Emergency Management*

⁵ <http://www.racer-systems.com/products/porter/index.phtml>

- Application. International Semantic Web Conference, Athens, GA, USA, pp. 959-966, (2006)
10. Adriana Lopez, Maria J. Somodevilla, Darnes Vilarino, Ivo H. Pineda and Concepcion P. de Celis: Toponym Disambiguation by Ontology in Spanish: Geographical proximity between place names in the same context. In: AISS: Advances in Information Sciences and Service Sciences, Vol. 4, No. 1, pp. 282-289 (2012)
 11. GeoWordNet: <http://geowordnet.semanticmatching.org/>
 12. GeoNames: <http://www.geonames.org/>
 13. Tripod: <http://www.research-projects.uzh.ch/p8052.htm>
 14. Buscaldi D.: Toponym disambiguation in information retrieval. *Procesamiento del Lenguaje Natural*, [46]:125–126 (2011)
 15. Budak Arpinar I., Amit Sheth, Cartic Ramakrishnan, E. Lynn Usery, Molly Azami & Mei-Po Kwan: Geospatial Ontology Development and Semantic Analytics. In: *Handbook of Geographic Information Science*, Eds: J. P. Wilson and A. S. Fotheringham, Blackwell Publishing (2004)
 16. Hugo Gonçalo Oliveira, Diana Santos, Paulo Gomes & Nuno Seco: PAPER: A lexical ontology for Portuguese. Springer-Verlag Berlin, Heidelberg (2008)
 17. Fangling Jiang & Wenjun Wang: Research on Chinese Toponym Ontology model. In: 2010 International Conference of Information Science and Management Engineering, (2010)

Base de Conocimiento de los procesos de desarrollo de software a través de un modelo de un sistema de gestión del conocimiento

Sergio Ruiz-Castilla¹, Yulia Ledeneva² y Rocío Morales¹

¹ Universidad Popular Autónoma del Estado de Puebla,
21 Sur 1103, Barrio Santiago, C.P. 72410, Puebla, México
jsrgioruiz@gmail.com, mariadelrocio.morales@upaep.mx

² Universidad Autónoma del Estado de México,
Unidad Académica Profesional Tianguistenco,
Paraje el Tejocote San Pedro Tlantizapan, 52600, Estado de México
yledeneva@yahoo.com

Resumen En la industria del software las organizaciones desarrollan proyectos de software por desarrolladores expertos con conocimiento individual, capitalizando conocimiento colectivo. En este artículo, se propone un modelo de un sistema de gestión del conocimiento para los procesos de desarrollo de software en México, para transferir el conocimiento explícito de los desarrolladores expertos a los no expertos. El sistema de gestión del conocimiento propuesta permite la captura y almacenamiento de activos de conocimiento en una base de conocimiento para su modelado y validación; por otro lado, se plantea la consulta de conocimiento por parte de los desarrolladores no expertos para la transferencia y reutilización del conocimiento. Se acentúan en el proceso de la transición del conocimiento tácito a explícito y culturalizar la voluntad de aportar conocimiento. El objetivo de este artículo consiste en desarrollar una aplicación para un sistema de gestión del conocimiento para la gestión y la consulta del conocimiento.

Palabras claves: Conocimiento tácito, conocimiento explícito, gestión del conocimiento, procesos de desarrollo de software, base de conocimiento.

1. Introducción

Al desarrollar proyectos de software se genera experiencia en los desarrolladores que se capitaliza en las mentes de los desarrolladores como conocimiento individual y en la organización se forja conocimiento colectivo que se debería de compartir para evitar que se tenga que gestionar en cada nuevo proyecto y organización.

El problema que se plantea en este artículo es como transferir el conocimiento de gestión de los Procesos de Desarrollo de Software (PDS) de los desarrolladores expertos y los no expertos. El conocimiento tácito en la mente de los desarrolladores expertos se traduce a conocimiento explícito para poder almacenarse en una base de conocimiento.

Los proyectos de software requieren de una serie de procesos de desarrollo que en poca o gran medida se repiten en otros proyectos, bajo el mismo u otro contexto. La ingeniería de software plantea la reutilización de código, pero es necesario evaluar la reutilización de conocimiento de los PDS.

Se propone un modelo de un Sistema de Gestión de Conocimiento (SGC) que permite la obtención del conocimiento de los PDS de los desarrolladores expertos para ser almacenado en una Base de Conocimiento (BC). Por lo anterior, se desarrolla una herramienta de software que permite el almacenamiento y consulta de activos de conocimiento con ayuda de un motor de búsqueda usando consultas en lenguaje natural por los desarrolladores no expertos.

Cuando el líder de proyectos de software debe desarrollar un proyecto, gestiona los requerimientos del proyecto y los almacena en un banco de requerimientos. Después diseña el software basado en los requerimientos generando experiencia en los desarrolladores de esos requerimientos. Cuando se inicia un nuevo proyecto, un bajo o alto porcentaje de esos requerimientos se repetirán en el mismo o diferente contexto.

La organización cada vez será más experta aprovechando el conocimiento colectivo de su capital humano, sin embargo existirán miembros que se incorporarán a la organización que carecen de los conocimientos que tiene la organización. Tomando en cuenta también los desarrolladores que deciden irse de la organización, que se marchan con el conocimiento tácito consigo mismo.

En este artículo, se plantean el siguiente objetivo: diseñar un modelo de un Sistema de Gestión del Conocimiento que permita la obtención y almacenamiento de activos de conocimiento de los Procesos de Desarrollo de Software en una Base de Conocimiento para transferir y reutilizar dicho conocimiento parcial o completamente en nuevos proyectos de software.

2. Conocimiento

El conocimiento es una mezcla fluida de experiencia estructurada, valores, información contextual e internalización experta que proporciona un marco para la evaluación e incorporación de nuevas experiencias e información. Se origina y se aplica en la mente de los conocedores. En las organizaciones con frecuencia no solo queda arraigado en documentos o bases de datos, sino también en las rutinas procesos, prácticas y normas institucionales [1].

Según Hessen [2] el conocimiento puede definirse como una transferencia de las propiedades del objeto al sujeto. Considerando que cada sujeto puede tener una diferente perspectiva del objeto. A medida que se conoce algo se logra cierto nivel de abstracción, por lo tanto a mayor nivel de abstracción mayor conocimiento se tiene del objeto.

Por otro lado, Alonso *et. al* [3] determina que el conocimiento es el conjunto completo de datos e información que se usa en la práctica para realizar ciertas acciones y crear nueva información.

2.1. Contexto del conocimiento

Los individuos tienen conocimiento individual que se incrementa día a día gracias a sus experiencias. El conocimiento individual se capitaliza en el grupo de trabajo donde está integrado. El conocimiento puede capitalizarse en las organizaciones de un consorcio o en un corporativo. Finalmente, el conocimiento de todas las organizaciones se capitaliza como el conocimiento de un país entero como se muestra en la Figura 1.

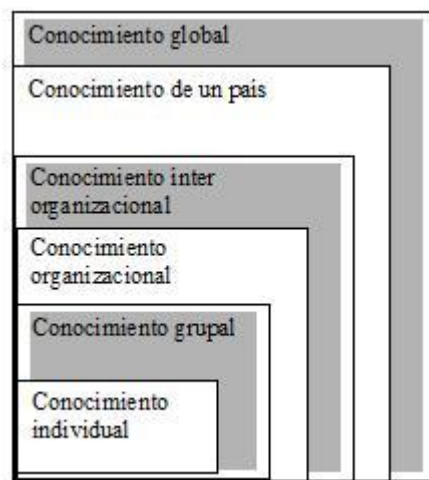


Figura 1. Dimensiones del conocimiento [4].

La propia ISO ocupada en la importancia que tiene la gestión del conocimiento en el apartado 12207 asevera que los recursos humanos de la organización son los encargados de asegurarse que el conocimiento, la información y las habilidades individuales se deben almacenar, reutilizar y mejorar con el tiempo [5]. Es decir, se sigue que las experiencias transformadas en conocimiento se gestionen como activos de la organización que tuvo un costo para generarse y que tiene un alto valor al aplicarse. Si bien el conocimiento no es tangible no implica que no exista, además tiene un valor e incluso se comercializa Davenport y Prusak manejan que la comercialización del conocimiento es una actividad en las organizaciones en forma de cursos y talleres de capacitación y en asesorías de consultores, el precio del conocimiento se establece cuando el proveedor le pone un precio y el cliente está dispuesto a pagar, considerando el beneficio que espera obtener con el nuevo conocimiento [1].

La gestión del conocimiento es una nueva disciplina que estudia las herramientas, técnicas y procedimientos para aprovechar al máximo potencial de los recursos humanos de la organización [6]. Se deberá considerar la experiencia, pericia y el aprendizaje para lograr una ventaja competitiva.

2.2. Conocimiento tácito

El conocimiento se puede clasificar en conocimiento tácito y conocimiento explícito. El conocimiento tácito es aquel que reside en la mente de los individuos y se caracteriza por ser difícil de codificar, formalizar y transmitir [6]. Por tanto, los expertos son los que poseen el conocimiento y le llevan consigo mismo. La propia complejidad de la transferencia del conocimiento tácito puede ser la razón por la cual no se documenta el conocimiento en las organizaciones.

2.3. Conocimiento explícito

El conocimiento explícito como se refiere Gómez y Suárez [6] es el conocimiento que se puede transmitir directamente mediante el lenguaje formal y en forma sistemática. También podemos encontrar este conocimiento en los libros. Es el conocimiento más abundante en forma de textos en las bibliotecas, enciclopedias, páginas Web, y otros medios; y es que tiene mayor transferencia debido a que está documentado y disponible (ver la Figura 2).



Figura 2. El ciclo conocimiento implícito – explícito [8].

3. Sistema de gestión del conocimiento

Un sistema de gestión del conocimiento tiene la misión de coleccionar, almacenar y permitir la consulta del conocimiento en una organización o comunidad con el fin de capitalizar el conocimiento como estrategia competitiva.

3.1. Base de conocimiento

Para almacenar el conocimiento es necesario un repositorio de archivos de los contenidos de los activos de conocimiento. Se usa una base de datos para los datos (metadatos) de los activos de conocimiento. Los anteriores pueden ser textos, aunque pueden contener imágenes. Pueden contener también archivos de sonido o videos. La Figura 3 muestra la arquitectura de la base de conocimiento. Cada activo de conocimiento tiene dos elementos:

- Los datos de los activo de conocimiento.
- Las carpetas que contienen los archivos de los activos de conocimiento clasificados.
- Los datos de los activos de conocimiento tienen la función e índices para el buscador de conocimiento.
- Las carpetas contenedoras de archivos de los activos de conocimiento, tienen la función de almacenar los archivos y su límite es la capacidad del disco duro u otros dispositivos de almacenamiento.

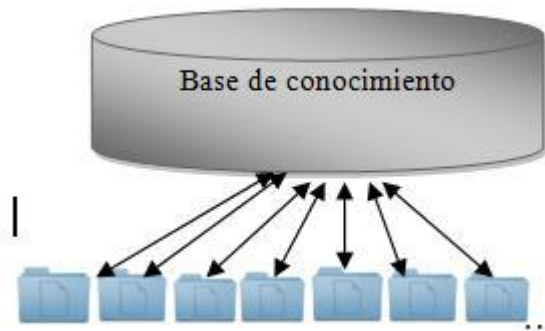


Figura 3. Propuesta de la arquitectura de la base de conocimiento.

El contenido de la base de conocimiento está constituido por el conjunto de archivos de activos de conocimiento:

$$CBC = CAAC1 + CAAC2 + CAAC3 + \dots \quad (1)$$

donde CBC es el contenido de la base de conocimiento y CAACn es el contenido de los archivos de los activos de conocimiento.

4. Administración del conocimiento

Es una actividad en el campo de actividades profesionales y empresariales que se centra en crear, recopilar, organizar y difundir el conocimiento de una organización, en lugar de la información o los datos [7].

La OCDE (Organización para la Cooperación y el Desarrollo Económico) a través del CIIE (Centro para la Investigación y la Innovación Educativa) define un modelo lineal (ver Figura 4) que consiste en 7 etapas [9]:



Figura 4. Modelo lineal de 7 pasos [9].

1. **Producción.** Se refiere a la producción de conocimiento de los individuos y de las organizaciones.
2. **Validación.** Se centra en que el conocimiento se haya aplicado de preferencia varias veces con resultados exitosos.
3. **Cotejo.** Se verifica que el conocimiento que se dispone realmente coincide con el conocimiento validado.
4. **Difusión.** Se determina cuáles son los medios a través de los cuales se hará la publicación e intercambio de conocimiento entre otros miembros de la organización, red o sociedad.
5. **Adopción.** Se motiva a los elementos de la organización para que abandone viejas prácticas y use el nuevo conocimiento que se le ha proporcionado.
6. **Implementación.** La mayoría de elementos de la organización se adoptan el conocimiento y existen las condiciones para dicha adopción.
7. **Institucionalización.** Se logran que la implementación del conocimiento sea algo normal. Se logran que sea una cultura de la organización en bien de todos [9].

5. Modelo propuesto

El modelo propuesto se muestra en la Figura 5. El módulo I, la Gestión de Proyectos (GP), se repite en cada proyecto que se desarrolla, de acuerdo a la metodología elegida y a los estándares internacionales para el desarrollo del software. Se repite el proceso, más no el proyecto, por lo que cada proyecto genera conocimiento nuevo. El equipo de desarrollo aplica conocimiento individual y colectivo para llevar a cabo cada tarea del proyecto. Cuando se finaliza un proyecto,

nacen nuevas experiencias que son conocimiento tácito adquirido. A mayor número de experiencias mayor conocimiento tácito individual y colectivo.

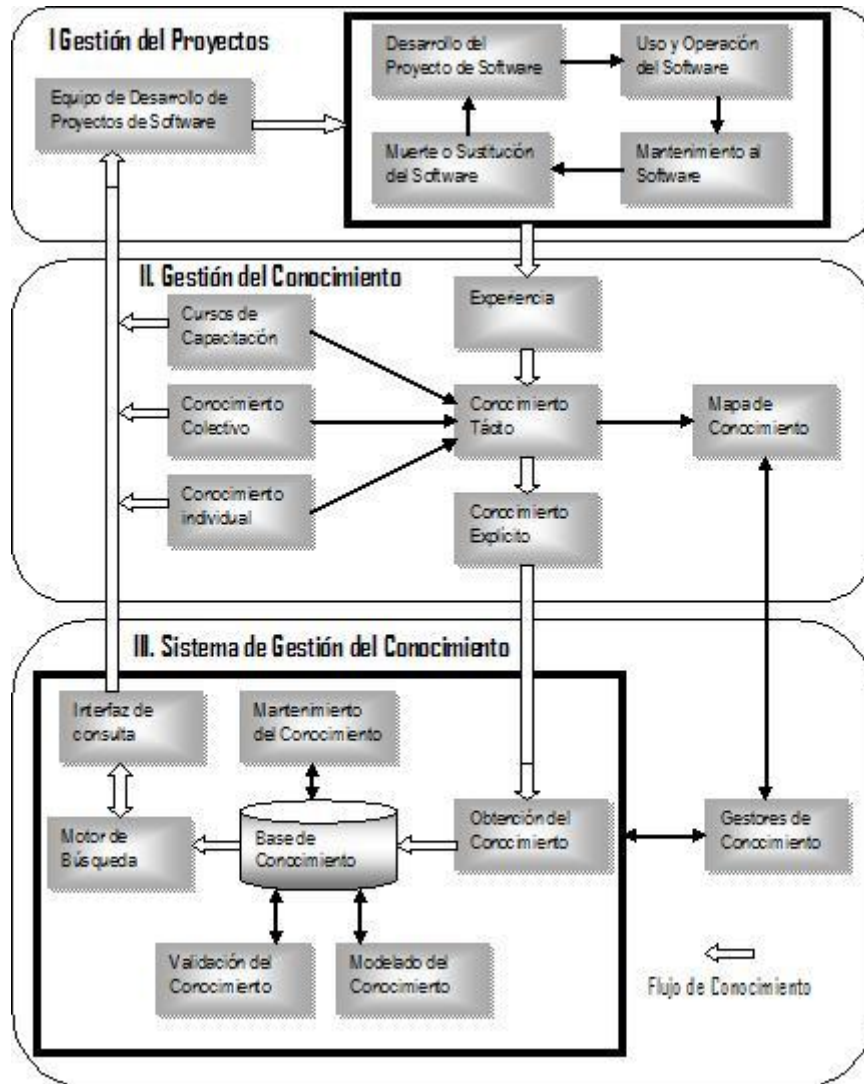


Figura 2. Propuesta del modelo del sistema de gestión del conocimiento para los procesos de desarrollo de software.

En el módulo II de la Figura 5 se muestra el proceso de la Gestión del Conocimiento (GC) referido a la tarea de generación, almacenamiento y uso del conocimiento. El conocimiento tácito se acumula de las experiencias y otras fuentes como, por ejemplo, la capacitación. Para almacenar el conocimiento es necesario

convertirlo del conocimiento tácito a explícito. La gestión del conocimiento es una tarea más que una herramienta tecnológica. Se requiere de cultura de gestión del conocimiento, que determina la importancia y la necesidad de capturar el conocimiento, almacenarse y disponerse para su uso a todos los desarrolladores.

Por último, en el módulo III se muestra el Sistema de Gestión del Conocimiento (SGC) que es la plataforma que se propone para la GC. El SGC facilita la obtención, almacenamiento y uso del conocimiento del conocimiento generado de la ejecución de los procesos al desarrollar el software. Contiene una BC que almacena y permite consultar conocimiento a través de un motor de consulta. La base de conocimiento que se plantea no es una base de conocimiento de reglas, sino más bien una especie de repositorio de activos de conocimiento. Es preciso además que el conocimiento se modele y se valide con el fin de mantener en la BC solo conocimiento de calidad.

En el modelo propuesto (ver la Figura 5) se puede observar una flecha gruesa del flujo del conocimiento que además tiene un ciclo que se repite acumulando más conocimiento organizacional. Una organización con la puesta en práctica del modelo le permite acumular conocimiento a través del tiempo y de explotarse de forma adecuada puede lograr mayor productividad y competitividad. El conocimiento que se administra con el SGC puede ser de carácter privado con acceso restringido o bien puede ser abierto para una comunidad, de acuerdo al tipo de conocimiento que se trate y de las políticas de la organización.

La BC es un conjunto de documentos que forman un compendio de Activos de Conocimiento (AC) que el motor de consultas usará para recuperar el conocimiento consultado por los desarrolladores no expertos. El potencial de la herramienta puede centrarse en el motor de búsqueda, considerando que el conocimiento se presenta como información no estructurada [10-13].

6. Implementación y validación del modelo

Una aplicación de software permite la gestión del conocimiento. El conocimiento tácito se obtiene de los desarrolladores expertos como se muestra en la Figura 6. Los gestores del conocimiento son los que se encargan de obtener y convertir el conocimiento tácito en explícito. Una de las formas para obtener el conocimiento es a través de relatos de las experiencias. Otra forma es cuando el desarrollador experto escribe un texto describiendo el conocimiento adquirido. Otros ejemplos pueden ser experiencias, prácticas, lecciones aprendidas, problemas y soluciones presentados en forma de archivos.

Los textos son activos de conocimiento de la organización que son integrados a la base de conocimiento como archivos de textos (ver la Figura 6). Es preciso catalogar cada activo de conocimiento para su clasificación dentro de la BC. Una vez almacenado el conocimiento se podrá modelar revisando el formato del texto y depurar errores de texto. El software permite que los usuarios del conocimiento califiquen el conocimiento en función de la utilidad obtenida, la validación la hacen entonces los propios usuarios. Cuando un activo de conocimiento obtiene calificaciones muy bajas será eliminado de forma automática. El conocimiento almacenado se puede consultar mediante el motor de consulta, mediante una palabra o

frase del activo de conocimiento que necesita el usuario del conocimiento (ver la Figura 7 y 8).

Es posible introducir conocimiento en formato explícito, en formato de texto que se almacenó en la base de conocimientos. El conocimiento corresponde a temas sobre los procesos de software. Los activos de conocimiento pueden revisarse en su contenido y también para comprobar su valor. Los usuarios del conocimiento pueden acceder y hacer consultas de conocimiento usando una palabra o una frase completa. La herramienta muestra un conjunto de activos de conocimiento disponibles en la base de conocimiento. El usuario del conocimiento elige algún activo de conocimiento y puede leer su contenido. Después de la consulta puede asignar una calificación con el fin de obtener un ranqueo del activo de conocimiento.

Grabar el componente

Núm. de Componente: 8 Tipo de proceso: 4 4 Calendización del proyecto

Tipo de Conocimiento: 6 6 Caso de éxito Desarrollador experto: 5 5 José José González Hernández

Resumen del componente de conocimiento:

Se consideraron factores del contexto para hacer el calendario del proyecto. Se manejó una holgura media para no castigar el costo del proyecto. Se logró un ajuste entre los factores contextuales y la holgura. La fecha de culminación y la planeada tuvo una diferencia de solo un día.

Contenido del componente de conocimiento:

La administración de software abarca la planeación, calendarización, administración de negocios, manejo del personal, estimación de los costos de software y la administración de calidad.

En este artículo se cubre la planeación y la calendarización de proyectos de software. Muchas son las causas para el fracaso de proyectos de software, se pueden mencionar: entrega tardía, no fiable, costo superior al estimado, características de ejecución pobres. Muchas veces la falla estaba en el enfoque de administración utilizado.

La necesidad de administrar es una distinción importante entre un desarrollo profesional de software y la programación no profesional. La administración de proyectos de software es necesaria debido a que la ingeniería de software profesional siempre está sujeta a restricciones de presupuesto y calendarización; a las que debe ajustarse la organización que desarrolla el software. El trabajo del

Figura 3. Incorporación de un activo de conocimiento.

Consulta de conocimiento

Introducir la(s) palabra(s) del conocimiento a buscar: estimación de costos Buscar...

	IdComponenteDeC	IdTipoDeProceso	IdTipoDeConocim	IDDesarrolladorExp	cdcResumen	cdcEstadoDeMode	cdcEstadoDeEval	cdcCalificacion
▶	4	3	4	2	Estimación de costos ...			
*								

Figura 4. Consulta de activos de conocimiento.

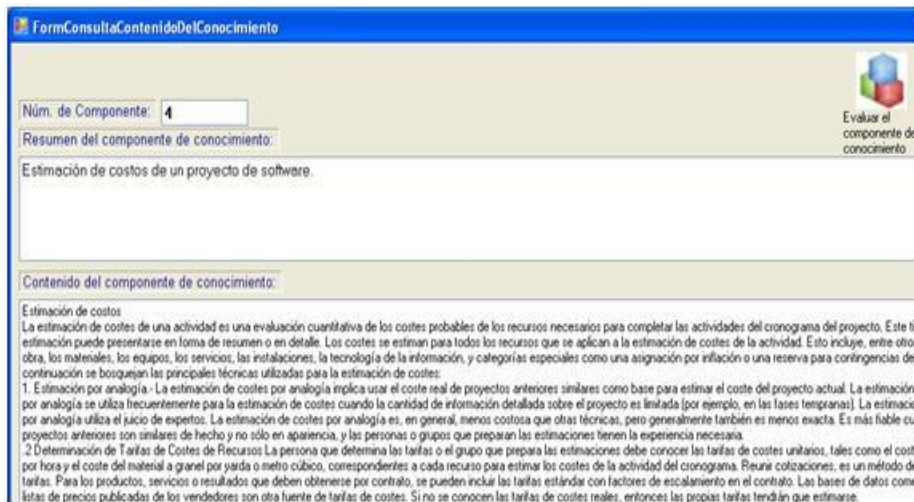


Figura 5. Visualización del activo de conocimiento.

7. Conclusiones

El conocimiento es un activo intangible en las organizaciones, sin embargo su valor en la industria del software es el activo más valioso. El conocimiento está en mente de los trabajadores, por lo que es necesario almacenarlo y disponerlo en una plataforma que permita su consulta en nuevos proyectos de desarrollo de software. A lo largo del tiempo los desarrolladores pueden aportar conocimiento que se acumula como conocimiento organizacional, bajo la premisa que se irán los trabajadores, pero el conocimiento se queda en la base de conocimiento. Una organización puede tener más productividad y competitividad en medida que el conocimiento que se gestiona. Es necesaria una cultura de la gestión del conocimiento para lograr su aportación, almacenamiento y utilización, bajo el convencimiento que es una medida que beneficia a toda la industria del software.

Referencias

1. Davenport, T. y Prusak, L.: Cómo las organizaciones manejan lo que saben. Buenos Aires: McGraw Hill. pp. 6 (2001)
2. Hessen, J.: Teoría del conocimiento. México: Época. pp. 20 (2008)
3. Alonso B. Amparo, Guijarro B. Bertha, Lozano T. Adolfo, Palma M. José Tomás, Taboada I. Ma. Jesús: Ingeniería del Conocimiento Aspectos Metodológicos, España: Pearson Prentice Hall. pp. 18 (2004)
4. Nonaka. I. Y Takeuchi, H.: La organización creadora del conocimiento. México: Oxford University Press (1999)
5. IEEE. IEEE/EIA 12207.1-1997. Industry Implementation of ISO/IEC 12207:1995 – Standard for Information Technology – Software Life Cycle Process (1997)

6. Gómez, A. Y Suárez, C.: *Sistemas de Información, Herramientas prácticas para la gestión empresarial*, (2a. ed.). México: Alfaomega Ra-Ma (2007)
7. Barnes S.: *Sistemas de Gestión del Conocimiento Teoría y Práctica*. España: Thompson Editores (2002)
8. Nonaka: *The Knowledge Creating Company*. Harvard Business Review. November – December 1991. Págs. 96-104 (1991)
9. OCDE y la Sociedad del Conocimiento FLACSO-México. *La administración del conocimiento en la sociedad del conocimiento*. Mayol Ediciones: Colombia (2006)
10. García-Hernández, R.A., Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A.: *A Fast Algorithm to Find All the Maximal Frequent Sequences in a Text*, In: Sanfeliu, A., Martínez-Trinidad, J. F., Carrasco-Ochoa, J. A. (eds.) CIARP 2004. LNCS vol. 3287, pp. 478-486. Springer-Verlag (2004)
11. García-Hernández, R.A., Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A.: *A New Algorithm for Fast Discovery of Maximal Sequential Patterns in a Document Collection*. In: Gelbukh, A. (ed.) CICLing 2006, LNCS vol. 3878, pp. 514–523, Springer-Verlag (2006)
12. Sidorov, G.: *Lemmatization in automatized system for compilation of personal style dictionaries of literary writers*. In: “Word of Dostoyevsky”, Russian Academy of Sciences, pp. 266-300, (1996)
13. Gelbukh, A. and Sidorov, G.: *Approach to construction of automatic morphological analysis systems for inflective languages with little effort*. Lecture Notes in Computer Science, N 2588, Springer-Verlag, pp. 215–220 (2003)
Gelbukh, A. and Sidorov, G.: *Analizador morfológico disponible: un recurso importante para PLN en español*. En: *Memorias de talleres del congreso internacional Iberamia-2004*, Puebla, México, noviembre 2004, pp. 209-216 (2004)

Modelo para diseño de perfiles de conocimiento: una aplicación en la industria generadora de energía eléctrica

María de Jesús Velázquez Mendoza, Oscar Mario Rodríguez-Elías,
César Enrique Rose Gómez y Sonia Regina Meneses Mendoza

División de Estudios de Posgrado e Investigación, Instituto Tecnológico de Hermosillo,
Av. Tecnológico S/N, Hermosillo, Sonora, 83170, México
{velazqu,omrodriguez,crose,so_meneses}@ith.mx

Resumen Dentro de las organizaciones, el capital intelectual está integrado por el capital humano, capital relacional, capital organizativo y capital tecnológico. En la actualidad, este valor intangible regularmente asocia el capital humano al capital relacional logrando estructurar un perfil de conocimiento limitado al contexto funcional y operativo. La integración del capital organizativo y tecnológico a dicha estructura de conocimiento posibilita la generación de una sólida base de ventajas competitivas para dichas organizaciones. En este trabajo se presenta una propuesta ontológica que permite estructurar el perfil de conocimiento de acuerdo a las prioridades de la organización. Se sigue una metodología para capturar y estructurar la base de conocimiento adecuada a las estrategias de negocio, así como la base de conocimiento adquirida por el capital humano. Ambas fuentes son la base retroalimentadora de la ontología propuesta para la inferencia de conocimiento acorde a las necesidades propias de la empresa.

Palabras clave: Competencias de conocimiento, capital intelectual, capacidades individuales y organizativas.

1. Introducción

Desde la perspectiva del conocimiento, el capital intelectual es considerado como el activo esencial del proceso organizacional, su contenido y naturaleza se centra en un lenguaje para pensar, hablar y hacer algo sobre los elementos conductores de los beneficios futuros de una compañía. Elementos como conocimiento, información, propiedad y experiencia figuran como material intelectual susceptible de generar riqueza, indudablemente esto parte del buen uso de los recursos y capacidades centrados en el ser humano.

En la actualidad la globalización marca para las organizaciones contextos cambiantes e inciertos en donde el éxito de las mismas depende fundamentalmente de la capacidad de adaptabilidad de los recursos con que estas cuentan [1]. Algunos procesos claves son la comprensión, creación de nuevas ideas, solución de problemas, toma de decisiones y acciones para el logro de los resultados deseados, esto lo integra el modelo ICAS KM (Sistema Inteligente de Adaptación Compleja para la GC) [2]. No obstante, las tecnologías de información (TI) y la gestión del recurso humano son dos variables que influyen directamente en el rendimiento de las organizaciones, por lo

que, las empresas que centran sus estrategias de TI en el uso estratégico de las mismas, obtienen un efecto favorable en el crecimiento o rentabilidad del negocio [3].

En este trabajo se expone una propuesta para el diseño de perfiles de conocimiento alineadas a las estrategias de negocio. Para ello, se sigue una metodología que nos permite capturar y estructurar la base de conocimiento de la organización, así como el conocimiento contenido en los elementos que interactúan en los procesos clave. Como último punto se plantea la creación de una ontología para su implementación en proyectos de gestión de recurso humano.

2. Antecedentes

Algunos autores expresan la necesidad de generar perfiles de conocimiento para la empresa en donde se integren las necesidades de conocimiento de los sectores académico, industrial y gobierno [4] y establecen principios organizadores de perfiles de conocimiento para la continuidad del conocimiento en las empresas [5], algunos de ellos son: 1) los perfiles de conocimiento sólo deben contener conocimiento operativo crítico, 2) los titulares de los cargos necesitan de un medio estructurado para la identificación de su conocimiento operativo crítico, 3) la naturaleza de los términos contractuales de los empleados incide sobre sus perfiles, 4) el contenido del perfil debe ser fácil de comprender y de acceso interactivo, y debe encerrar significado tanto para los titulares como los sucesores, 5) el acceso a ciertos aspectos del perfil de conocimiento debe ser restringido, entre otros.

Para tal efecto se ha tomado como Capital Intelectual todas aquellas prácticas que integran diferentes activos intangibles dinamizados por una estrategia basada en flujos de conocimiento o en actividades intelectuales dentro de la organización, esto para crear valor o riqueza cognitiva, compuesta por un conjunto de activos de naturaleza intangible o recursos y capacidades basados en conocimiento [6].

3. Alineación estratégica

Para este proyecto se propone integrar los recursos y capacidades contenidos en toda organización (Recurso Humano, Procesos, Fuentes de información, etc.) alineados hacia el objetivo de negocio con el propósito de hacer de esta un organismo más rentable. La figura 1, nos muestra el punto de partida al momento de generar un perfil de conocimiento, así como las áreas principales a considerar en la estrategia de negocio. Como puede observarse, el área contenedora de la fuente principal del capital intelectual está en la gestión del recurso humano, y es precisamente este elemento quien debe involucrar todas sus capacidades (GC) para el uso estratégico de los recursos que estén a su alcance (TI) en cualesquier actividad a desempeñar (GP). Esto permite garantizar el fortalecimiento constante del objetivo de negocio.

En la red semántica que se muestra en la figura 2 se representan dos perspectivas al momento de gestionar el conocimiento para su alineación estratégica dentro de la empresa: afín a la organización y afín al recurso humano. Desde el punto de vista de la organización, tenemos que un rol dentro de un proceso de trabajo, requiere de ciertas herramientas y conocimiento a un nivel óptimo para alcanzar los objetivos organi-

zacionales. Por otro lado, el recurso humano, representado por el empleado, se integra a un proceso de trabajo en donde juega un rol y cuenta con cierto nivel de conocimiento.

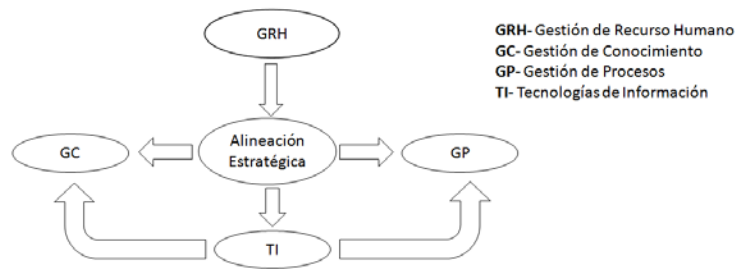


Fig. 1. Alineación estratégica del recurso humano.

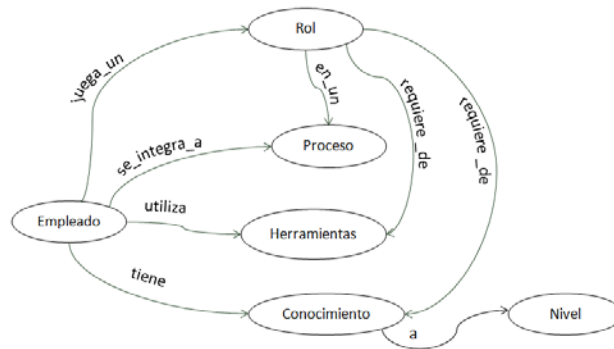


Fig. 2. Red semántica de la gestión de recursos y capacidades en la empresa.

Considerando estos enfoques visualizamos la necesidad de conceptualizar dichas relaciones, para ello hemos definimos como Perfil el “Conjunto de rasgos descriptivos que caracterizan la personalidad de un individuo” y como Perfil de Conocimiento al “Conjunto de rasgos estructurados que describen el conocimiento requerido, asociado a los recursos y capacidades, que permitan habilitar la generación dinámica de competencias laborales, competencias de los procesos clave y competencias distintivas que generen valor a la organización”.

4. Propuesta de un modelo de perfiles de conocimiento para una empresa generadora de electricidad

Durante el desarrollo de este trabajo se implementó una extensión de la metodología KoFI [7]. Esta ha sido diseñada para proponer soluciones basadas en la gestión de conocimiento, tiene un enfoque orientado al análisis de procesos específicos y flujos de conocimiento, sus principales etapas son: a) el modelado del conocimiento centrado en el proceso, b) el análisis del proceso (incluye la identificación de las fuentes,

tipos y flujos de conocimiento), y c) los problemas de flujo de conocimiento. En esta investigación nos centramos en las dos primeras etapas e incluimos una que nos permite generar un esquema para la representación del conocimiento.

4.1. Recopilación de información

En esta etapa se hizo un estudio de la información estratégica de la organización y se identificaron los procesos clave, las actividades realizadas fueron:

- Recopilación de información relacionada con las estrategias de negocio. Esto nos brindó un panorama general de la Misión, visión y política de la empresa.
- Recopilación de documentación de los procesos. Nos permitió esquematizar el mapa de procesos integrado e interrelacionarlos con las reglas de negocio y satisfacción del cliente al momento de generar un producto.
- Entrevistas a los principales actores de la organización. Las reuniones de trabajo fueron exclusivamente para analizar la estructura de los procesos y el conocimiento contenido en cada uno de ellos.
- Identificación de los procesos clave. El mapa de procesos nos arrojó como procesos clave: Gestión de Operación, Gestión de Mantenimiento y Medición, Análisis y Mejora.
- Delimitación de las áreas involucradas en los procesos clave. Enfocándonos en el proceso clave Gestión de operación se identificaron 6 cargos o puestos de trabajo con diferentes niveles de conocimiento.

4.2. Modelado de conocimiento con enfoque a proceso

En esta etapa, para el modelado del proceso de operación se utilizó la adaptación de [8: pp 88-99] sobre la técnica de modelado llamada Gráfica Rica [9]. Esta técnica de modelado nos facilitó la delimitación de las actividades, fuentes de información y herramientas asociadas a cada uno de los roles dentro del proceso.

Bajo dicha técnica, en las figuras 3 y 4 se muestra el proceso clave Operación de Ciclos Combinados, representado por una nube. En este proceso intervienen 6 áreas, correspondientes a 6 diferentes cargos organizacionales, su representación es por medio de uno o más actores (dependiendo del número de individuos que cubren dicho cargo), las líneas no dirigidas hacia la nube nos indican qué roles intervienen en dicho proceso. Los temas de conocimiento figuran en corchetes, si la línea se dirige del actor a los temas de conocimiento nos muestra que el rol debe tener ese conocimiento y utilizarlo en el proceso, si la línea tiene una dirección inversa, nos marca que el actor obtiene ese conocimiento durante su participación en el proceso. En cuanto las fuentes de conocimiento, se representan 3 tipos: sistemas de información (representado por un rombo), documentos (representado por un rectángulo) y herramientas de trabajo (representado por doble marco). Una línea dirigida hacia la nube nos señala que la fuente se utiliza en el proceso y una línea bidireccional que la fuente se modifica durante el proceso. Todo esto para la obtención de un producto final, energía eléctrica.

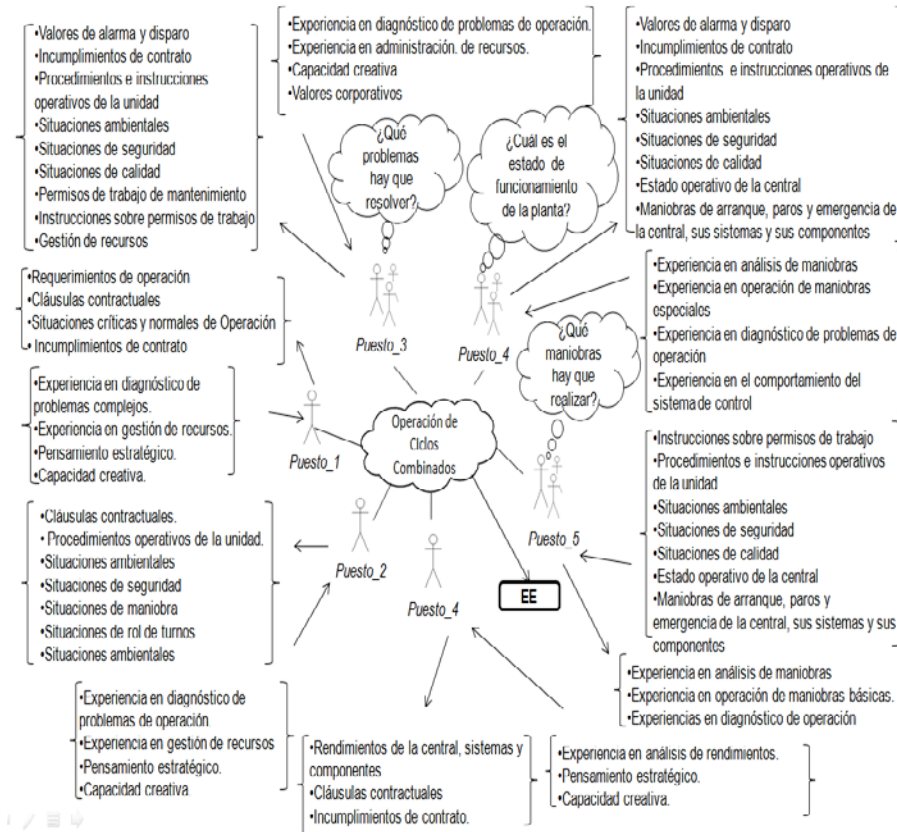


Fig. 3. Gráfica Rica del proceso de operación.

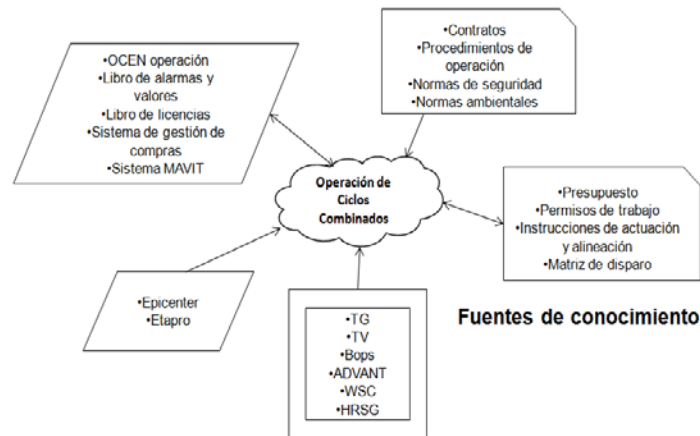


Fig. 4. Fuentes de conocimiento para el proceso de operación.

4.3. Análisis de flujo de conocimiento

Posteriormente se procedió a delimitar la base de conocimiento, para ello se realizó un profundo análisis del proceso que nos llevó a clasificar las fuentes y los temas de conocimiento, así como su ubicación (sistemas de información, documentos, personas clave, etc.) y las relaciones de todos sus elementos con las actividades principales del proceso.

1. **Identificación y clasificación de las fuentes de conocimiento.** Basándonos en la teoría de recursos y capacidades, y enfocándonos exclusivamente en los recursos, describimos a estos como los “activos de que dispone la empresa para llevar a cabo su estrategia competitiva” [10: p 122] mismos que pueden ser clasificados en tangibles (físicos y financieros) e intangibles (no humanos y humanos), en donde los no humanos son clasificados en los tecnológicos y organizativos [11]. Para esta empresa, la parte documental funge como la principal fuente de conocimiento para la función organizativa, en cuanto a las herramientas de trabajo ha sido necesario delimitar su enfoque, por lo que se realizó una redistribución de los recursos no humanos implementando el metamodelo de fuentes de conocimiento de [8: pp 61-63]. La clasificación de recursos para esta empresa son: empleado, documento, y herramientas; estas últimas clasificadas en informáticas, tecnológicas y manuales.
2. **Identificación y clasificación de áreas de conocimiento.** El aprendizaje organizacional es observado como una herramienta que le otorga un carácter dinámico o de flujo a la gestión del conocimiento, y puede ser visto desde diferentes enfoques [2: pp 4-5]. Nosotros tomamos la perspectiva de las tecnologías y procesos, e integramos las áreas de conocimiento que engloban al conjunto de tareas y recursos que apoyan al proceso. Las áreas de conocimiento para el proceso de operación (ciclos combinados) son: técnico operativo, gestión de operación, seguridad, calidad, formación de personal e idioma extranjero.
3. **Identificación de flujos de conocimiento.** Posteriormente nos centramos en ubicar la forma en que interactúan las áreas y fuentes de conocimiento dentro del proceso, así como las tareas que se deben realizar y las decisiones que se han de tomar. Para ello, ha sido necesario realizar su modelado. Se ha optado por la técnica de modelado BPMN (Business Process Model and Notation), esta fue desarrollado para proporcionar una forma estándar de representar procesos de negocio tanto para propósitos descriptivos de alto nivel como para procesos detallados con entornos de software orientados a procesos [12]. Además, es sumamente flexible al momento indicar fuentes de conocimiento, dando apertura a la inclusión de artefactos personalizados para la identificación de las fuentes de conocimiento involucradas en las diversas tareas. En la figura 5, se detalla en términos generales el proceso clave *Operación de Ciclos Combinados*. Cada carril corresponde a un puesto representado en la estructura organizacional de la empresa. Como puede observarse existen 3 cargos que operan directamente con las herramientas tecnológicas de la planta C3 (5 individuos), C4 (5 individuos) y C5 (6 individuos), distribuidos en 5 turnos de trabajo. Esta parte tiene una secuencia Ad Hoc ya que sus tareas pueden producirse en cualquier orden y frecuencia, es decir, dentro del proceso no existen decisiones obvias o un orden específico de ejecución de tareas. El resto de los cargos (que no

son menos importantes ya que deben estar alineados con la estrategia de negocio), tiene mayor interacción con tecnologías de información o herramientas informáticas y un menor grado de interacción con herramientas tecnológicas. Aun así, los requerimientos para estos últimos son de un alto nivel de conocimiento.

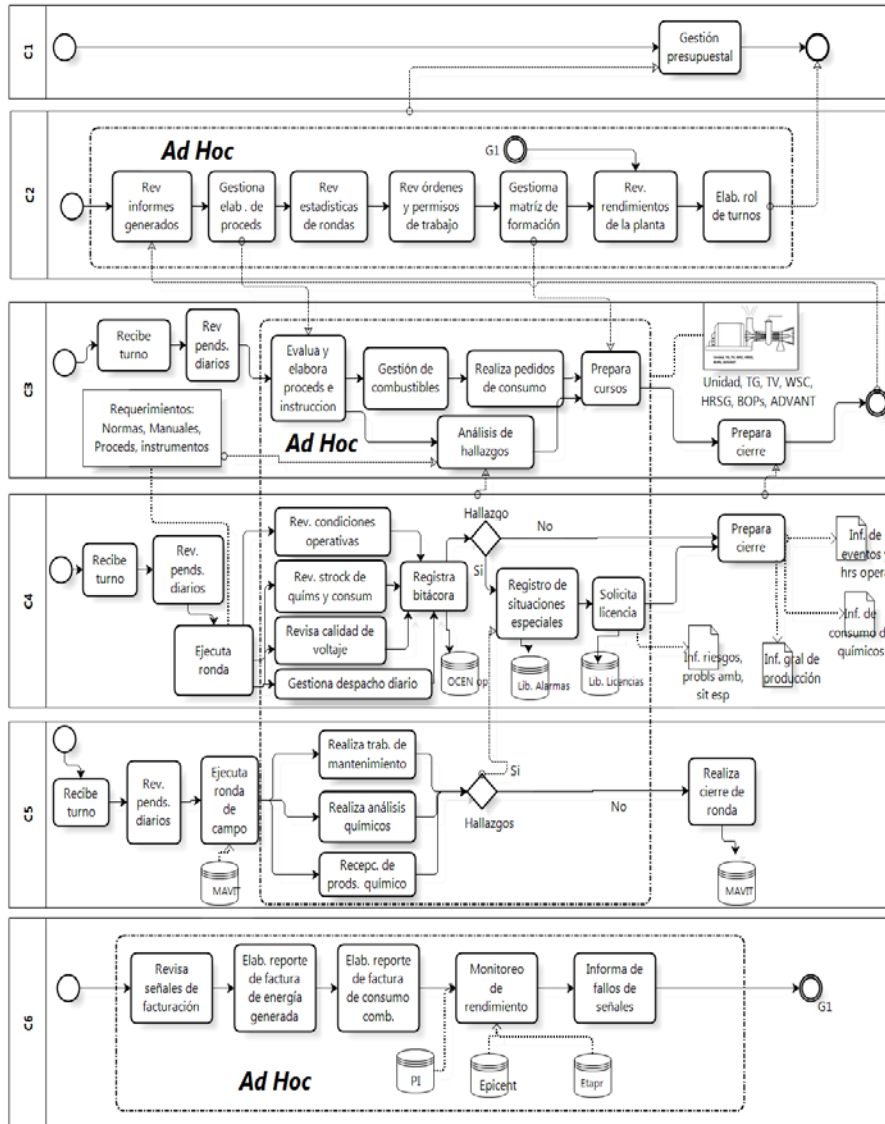


Fig. 5. Proceso clave Operación de Ciclos Combinados.

Análisis e identificación de herramientas y prácticas de KM. Las herramientas involucradas en los flujos de conocimiento dentro de este caso de estudio, son de

gran apoyo en las prácticas que ejecutan cada uno de los miembros de este equipo de trabajo. Su evaluación se realizó a través del marco de análisis de herramientas y tecnologías facilitadoras de flujos de conocimientos de [13], siendo detectadas 15 diferentes áreas de conocimiento para 3 roles específicos del proceso. Los niveles de conocimiento fueron delimitados según su alcance dentro del dominio en: 1) elemental, 2) medio, 3) avanzado y 4) experto. Es decir, la aplicación del esquema de clasificación dependió de los temas que debe dominar el individuo dentro de cada área de conocimiento. Una perspectiva gráfica de los niveles de conocimiento requerido para los cargos al momento de ejecutar un rol (formador, analista u operador) se muestra en la figura 6, o de forma más genérica se puede ver en la figura 7 los niveles de conocimiento por área, independientemente del rol que el cargo vaya a ejecutar.

Nivel de Conocimiento por Rol

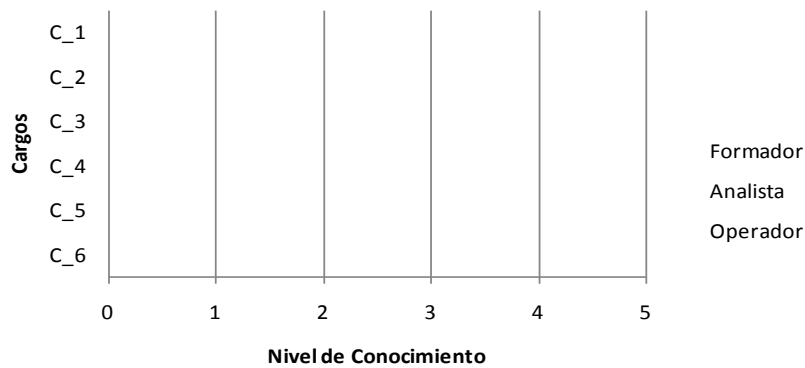


Fig. 6. Nivel de conocimiento requerido por rol en el proceso de Operación CC.

Niveles por Área de Conocimiento

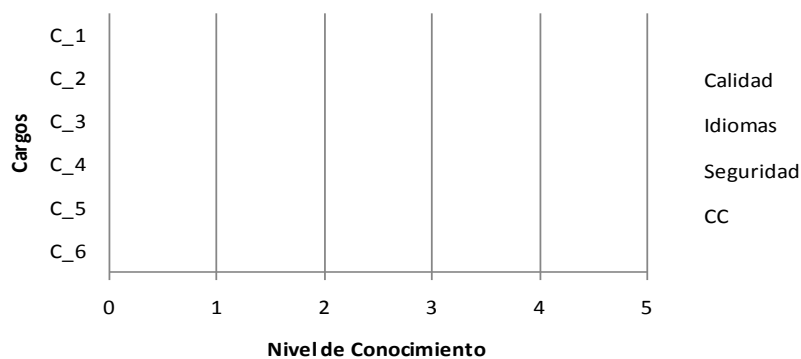


Fig. 7. Nivel de conocimiento requerido por área o tema de conocimiento en el proceso de Operación CC.

4.4. Representación ontológica del conocimiento

La base de conocimiento del proceso evaluado en este trabajo de investigación, se representa a través de la ontología de la figura 8.

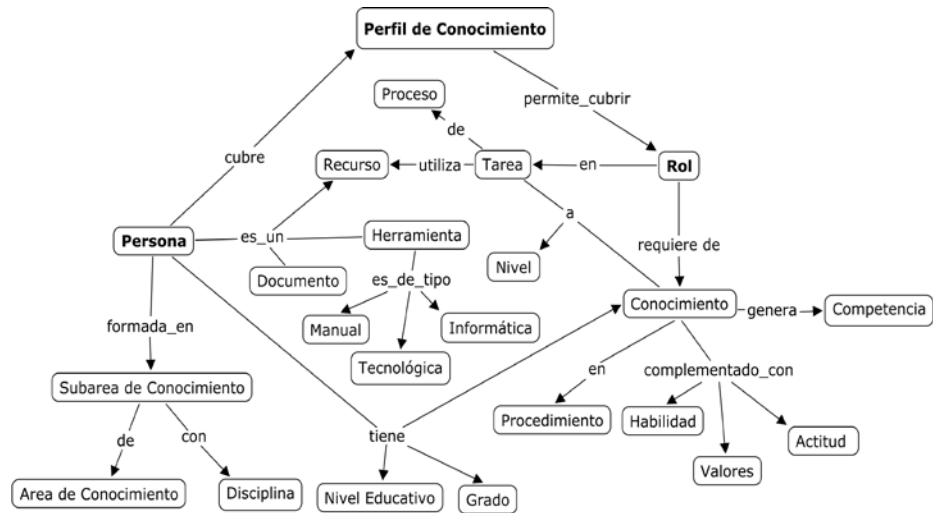


Fig. 6. Ontología perfiles de conocimiento.



Fig. 7. Vista parcial de la ontología perfiles de conocimiento.

Esta forma de representación es considerada un medio útil para su estructuración y documentación [14], ya que su analogía facilita la comunicación y el intercambio de información entre sus entidades al momento de inferir conocimiento. La conceptualización y sus relaciones se establecieron siguiendo el ciclo de vida de Methontology [15].

La ontología propuesta ha sido implementada en Protégé, con una muestra de datos con información acerca de los conocimientos óptimos para cada uno de los roles dentro del proceso clave, lo que nos ayudó a perfilar el conocimiento requerido para los cargos. Por otra parte, aplicando el mismo esquema ontológico, también se ha tomado una muestra contenedora de conocimiento para actores que cubren dichos cargos dentro de la organización, esto con el objeto de perfilar el conocimiento del individuo.

5. Conclusiones

El desarrollo de este proyecto ha enriquecido nuestra percepción acerca del capital intelectual en las organizaciones. Con ello, se logró estructurar el conocimiento del proceso clave e identificar los elementos que interactúan alineándolos hacia el objetivo de negocio. La implementación de la ontología, nos permitió demostrar que las relaciones que se establecen entre todos los elementos que interactúan en la organización, difícilmente pueden ser observadas a través de una asociación tácita. Por lo que, la buena gestión de los recursos a través de nuevas herramientas tecnológicas aplicadas a la administración de la base contenedora de conocimiento incrementa el potencial de los recursos facilitando el alcance de las ventajas competitivas de la empresa.

Actualmente se está desarrollando la interfaz por medio de la cual se explotará la ontología y la base de conocimientos, se espera que en la primera etapa de aplicación:

- (a) Unirla con el programa de capacitación y determinar qué conocimiento crítico requiere adquirir el personal involucrado en dicho proceso, y
- (b) Determinar el plan de carrera para los actores involucrados.

Visualizando que esta estructura tiene apertura para la toma de decisiones, también consideramos su futura aplicación para la integración de equipos de trabajo en proyectos especiales.

Agradecimientos. Agradecemos el apoyo otorgado mediante beca comisión de DGEST al primer autor (Oficio 513.2.2/0542/12). De igual forma el financiamiento parcial de PROMEP (Oficio 103.5/11/840) para el desarrollo de este proyecto.

Referencias

1. J. R. Morales Calderon and S. Salvador García: Dirección y Administración de Personal en el Siglo XXI - Actualidad y desafíos. Administración, pp. 135-152 (2006)
2. K. Dalkir: Knowledge Management in Theory and Practice. Burlington, MA, USA: Elsevier, pp. 1-356 (2005)
3. Y. Yang Chen and H. Ling Huang: The Knowledge Management Strategic Alignment Model (KMSAM) and Its Impact on Performance: An Empirical Examination. In:

- Knowledge Management, P. Virtanen and N. Helander, Eds. Vukovar, Croatia: In-Tech, pp. 25-34 (2010)
4. O. M. Rodríguez Elías and A. I. Martínez García: Hacia la definición de un esquema de clasificación de áreas de conocimiento para ingeniería de software en México. En: Avances en la Ciencia de la Computación: Memorias de los talleres del ENC 2006, A. Hernández Aguirre and J. L. Zechinelli, Eds. San Luis Potosí, México: pp. 47-52 (2006)
 5. H. Beazley, J. Boenisch, and D. Harden: Continuity Management — Preserving Corporate Knowledge and Productivity When Employees Leave. USA: John Wiley & Sons, pp. 1-269 (2002)
 6. E. Bueno, M. Paz, and C. Merino: Génesis, concepto y desarrollo del capital intelectual en la economía del conocimiento: Una reflexión sobre el Modelo Intellectus y sus aplicaciones. Estudios de Economía Aplicada, vol. 26, no. 2, pp. 43-64 (2008)
 7. O. M. Rodríguez Elías, C. E. Rose Gómez, A. Vizcaíno, and A. I. Martínez García: Integrating Current Practices and Information Systems in KM Initiatives. In: KMIS 2010 International Conference on Knowledge Management and Information Sharing, pp. 71-80 (2010)
 8. O. M. Rodríguez Elías: Metodología para el Diseño de Sistemas de Administración del Conocimiento: su aplicación en mantenimiento de software. Centro de Investigación Científica y de Educación de Ensenada (2007)
 9. A. Monk and S. Howard: The Rich Picture: A Tool for Reasoning About Work Context. Interaction, vol. 5, no. 2, pp. 21-30 (1998).
 10. J. Maroto: Estrategia de la visión de acción. 2nd ed. Madrid: ESIC Editorial, pp. 1-467 (2007)
 11. J. E. Navas and L. A. Guerras: La Dirección Estratégica de la Empresa. Madrid: Civitas, (2002)
 12. S. A. White and M. Derek: BPMN modeling and reference guide. Lighthouse. Future Strategies Inc (2008)
 13. O. M. Rodríguez Elías, A. I. Martínez García, A. Vizcaíno, J. Favela, and M. Piattini: A framework to analyze information systems as knowledge flow facilitators. Elsevier, pp. 481-498 (2007)
 14. A. Perez Soltero, M. Barcelo Valenzuela, G. Sanchez Schmitz, and O. M. Rodríguez Elías: A computer prototype to support knowledge audits in organizations. Knowledge and Process Management, vol. 16, no. 3, pp. 124-133 (2009)
 15. A. Gomez Pérez, M. Fernández López, and O. Corcho: Ontological Engineering. United States of America: Springer (2004)

Generación semiautomática de ontologías utilizando bases de datos relacionales como fuente primaria de información

David González-Marrón^{1,2}, Miguel González-Mendoza²
y Neil Hernández-Gress²

¹ Instituto Tecnológico de Pachuca, Carretera México-Pachuca Km 81.5, Pachuca, Hidalgo, México
dgonzalez@itpachuca.edu.mx

² Tecnológico de Monterrey, Carretera Lago de Guadalupe Km 2.5, Atizapán de Zaragoza, Edo. de México, México
{mgonza,ngress}@itesm.mx

Resumen Actualmente, existen muchos investigadores buscando automatizar la producción de ontologías, utilizando ingeniería inversa para la extracción de información contenida en las bases de datos, sin embargo debido a la continua evolución de estándares en las bases de datos relacionales, este proceso solo se ha podido realizar de manera parcial. Es por esto que muchas de las características en la definición de almacenamiento existentes en los modernos manejadores de bases de datos no son soportadas aún por la tecnología semántica. Otro factor importante es que una misma base de datos puede ser implementada con diferentes comandos de definición de datos por diferentes diseñadores, lo que ocasiona diferencias en las ontologías generadas. Este trabajo está enfocado principalmente a describir como pueden ser generadas las ontologías a partir de las bases de datos relacionales, los principales problemas para realizar esta actividad de manera automática, los lenguajes que pueden ser utilizados para realizar ésta actividad, y propone una alternativa para generar ontologías de manera semiautomática utilizando software libre desarrollado por la comunidad.

Palabras clave: Bases de datos heterogéneas, Web semántica, ontologías.

1. Introducción

La mayor parte de la información accesible en la web, es extraída de bases de datos utilizando programas especialmente diseñados. La cantidad de la información existente en estas bases de datos es más del 70 % de la información en la Web, esta información existente en las bases de datos es comúnmente conocida como la “deep web” [1] debido a que no es accesible a través de motores de búsqueda de propósito general. La web semántica está en búsqueda de mecanismos que permitan encontrar, compartir y combinar la información de la web más fácilmente [2].

EL uso de ontologías permiten incrementar la interoperabilidad entre los sistemas, al mismo tiempo permite el uso de la tecnología semántica para incrementar el po-

tencial de las consultas formuladas. En este trabajo un análisis de los mecanismos requeridos para producir ontologías automáticas es realizado, se mencionan los lenguajes y comandos mas utilizados para producir ontologías, adicionalmente un análisis de las ventajas y desventajas de explotación de información semántica para un usuario común es realizado, así como del proceso para producir ontologías. La web semántica busca la integración de archivos RDF, sin embargo debido a que la mayor parte de la información está almacenada en Bases de Datos, un mecanismo para interactuar con esta información debe ser implementado, uno de los métodos más utilizados por investigadores y desarrolladores es el mapeo de esquemas definidos en bases de datos en SQL a archivos en lenguajes OWL ó RDFS principalmente. En la siguiente tabla se muestra una relación entre SQL y la web semántica. Puede ser visto en la tabla en su nivel inferior que los datos pueden ser convertidos en archivos RDF (conformado por tripletas), en el nivel superior los triggers permiten describir las operaciones que se realizan al hacer operaciones de actualización de datos. Actualmente no existe una herramienta automática que genere reglas, principalmente por la complejidad de los triggers (disparadores). El estado del arte es la generación de ontologías básicas. Estas ontologías son nombradas por Sequeda et al [3] como “ontologías putativas”. Debido a que son ontologías simples que requieren ser validadas y complementadas por expertos de dominio.

Tabla 1. Niveles de equivalencia entre componentes de SQL y lenguajes web semánticos.

SQL	Semantic Web
TRIGGERS	RULES
CONSTRAINTS	OWL
TABLE DEFINITION	RDFS
RELATIONAL MODEL	RDF

1.1. Integración de bases de datos

La integración de bases de datos, es considerada como una de las tareas más difíciles en la web semántica principalmente por el gran número de variables involucradas en esta actividad, tales como: incompatibilidad de datos, distribución de datos, diferente significado semántico de los datos, mecanismos de seguridad, tareas de mapeo de las bases de datos, diferente representación y precisión de los datos. Este trabajo está más enfocado a lograr interoperabilidad en las bases de datos utilizando ontologías utilizando un enfoque semántico. Los avances más significativos en el área han sido realizados por el grupo OMG [11]. Un importante estudio de las principales herramientas existentes para generar mapeos automáticos es realizado por el grupo incubador del RDB2RDF [12].

1.2. Aproximación para acceder BD relacionales utilizando métodos semánticos

Para crear mapeos utilizando bases de datos relacionales, la primera opción es la creación de mapeos estáticos del esquema de las BD, este proceso se encuentra aún en desarrollo por diferentes investigadores, el segundo es un proceso semiautomático que

transforma la información a un formato semántico. A continuación se hace una breve descripción de cada método

Realización de mapeos estáticos. En este método, el mapeo es realizado sobre todo el esquema de la BD sin realizar copias físicas de la información, la estructura de las tablas de las BD relacionales y sus atributos son mapeados a sus correspondientes clases y propiedades en una ontología definida por el usuario. El esquema de BD relacionales es una tupla $R(U, D, \text{dom}, I)$. Donde : R es conocido como una relación definida como un conjunto de tuplas que tienen los mismos atributos , y una tupla usualmente representa información acerca de un objeto; U es un conjunto finito de atributos en una relación; D es el dominio de la BD; dom representa una función que mapea U a D , donde cada atributo A_i en U debe tener un valor válido en D_i ; I es un conjunto finito de restricciones de integridad, el cual restringe las instancias de datos almacenadas en la BD, dos diferentes tipos de integridad pueden ser aplicados (integridad de datos e integridad referencial). La siguiente figura muestra el proceso requerido para generar información a partir de una BD relacional que sea de utilidad para un motor de consultas semántico. Actualmente los resultados producidos por la tecnología no son satisfactorios, debido a que todavía se requiere un proceso de refinación por expertos en las áreas del dominio de discurso

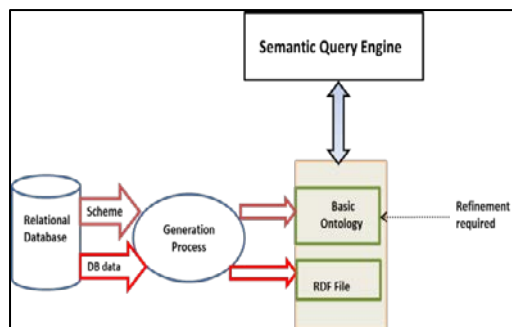


Fig. 1. Mapeo directo de bases de datos relacionales a datos semánticos.

Realización de conversión de las bases de datos (Wrapping Systems). Este método produce una conversión de la BD a un formato diferente, comúnmente conformado por tripletas de los siguientes tres elementos (Sujeto, Predicado y Objeto). Aquí la conversión produce un gran incremento de la información. Esta conversión generalmente es realizada en formato RDF, el cual es almacenado en repositorios que pueden ser explotados utilizando SPARQL. Diferentes herramientas existen para almacenar los archivos RDF generados de manera que los datos sean almacenados en una sola plataforma. EL proceso de creación de ontologías normalmente utiliza una ontología de dominio, la cual ayuda a producir información semántica, esta ontología de dominio si no es utilizada, produce normalmente tripletas en un formato común de RDF, en este caso se considera la premisa de un mundo cerrado “closed world assumption”, y los URIs “Uniform Resource Identifier” (identificadores de recursos uniformes) creados son muy simples, en cambio cuando se utiliza una ontología de dominio las URIs generadas incluyen información acerca del dominio utilizado.

2. Porqué utilizar la integración semántica

Los aspectos generales relacionados con las aplicaciones de las BD es que la integración semántica puede solucionar diferentes problemas que se presentan en las representaciones estructuradas, como la codificación de datos con más de una representación, es por eso que las aplicaciones deben resolver heterogeneidades con respecto a los esquemas y sus datos, la representación semántica permite la manipulación de datos y permite la transformación de datos entre diferentes esquemas de BD [13]. Esta tarea se relaciona principalmente con la conversión de datos para la explotación de consultas. Una conceptualización de explotación de consultas se muestra en la figura 2, en la cual un usuario solicita una consulta que requiere la integración de diferentes bases de datos, como la mayor parte de las veces los datos no son compatibles, el uso de la integración semántica es una propuesta a ser considerada.

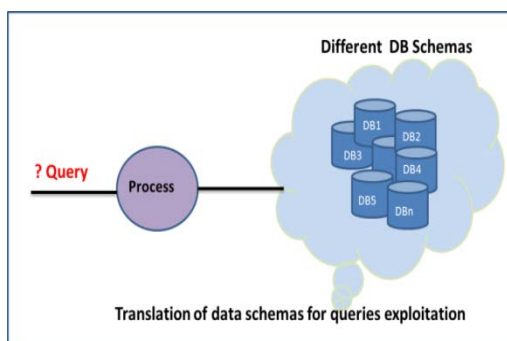


Fig. 2. Vista conceptual de explotación de consultas semánticas.

2.1. Alternativas para la integración de datos

Dos alternativas existen para lograr la integración de datos, la más común es la estática, utilizando ETL “Extracción, Transformación y Cargado (*Load*)”, la cual produce conversión de los datos de un área temporal a un área de representación final. La otra aproximación es la dinámica (virtualización de datos), en la cual los datos permanecen en su fuente de origen, y una vista conceptual de datos es producida tan pronto se necesita. El ETL tradicional es una variación del paradigma ETL, en el cual las fuentes de datos son usadas para soportar aplicaciones de negocios, y estas aplicaciones normalmente requieren formatos diferentes, archivos, estructuras y aún diferentes codificaciones. Normalmente existe una tendencia para normalizar datos antes de integrarlos a un sistema objetivo.

La virtualización de datos es contraria al método tradicional de extracción de datos de diferentes fuentes y de almacenamiento temporal de estos datos en un área de trabajo. Este método denominado federación de datos o virtualización de datos permite que las fuentes de datos permanezcan en sus localizaciones originales. La virtualización es soportada por una variedad de fuentes de datos nativas y proporciona vistas relacionales sin requerir que los datos sean extraídos de su fuente. Los consumidores de datos toman una vista de datos estructural y semánticamente consistente.

Otra aproximación novedosa a ser utilizada para integrar datos es la integración semántica, la cual es un área activa dentro del área de las bases de datos, integración de información y ontologías, permitiendo la interoperabilidad entre diferentes sistemas. La integración semántica involucra también técnicas de alineación con esquemas de bases de datos para responder consultas complejas que requieren la utilización de múltiples fuentes, aquí los datos son transformados a fin de poder utilizar la semántica de los datos, dos diferentes alternativas son de uso potencial. El primero es similar a la aproximación utilizada por ETL, pero utilizando una organización semántica de los datos y replicándolos. El segundo está relacionado con la virtualización de los datos, creando archivos de mapeo necesarios para localizar los datos.

2.2. Porqué la integración semántica

La necesidad de uso de información semántica, es causada debido a la diversidad de representación de datos y la información que normalmente se encuentra interrelacionada entre sí, es importante mencionar que debido a que la información no se encuentra contextualizada, la mayor parte de los desarrolladores podrían no pensar en los datos en sí, sino en la estructura de los datos, tal como: tipos de datos, esquemas, formatos de archivos, construcciones de BD y otras estructuras que no están relacionadas directamente con la información.

3. Problemática para modelar ontologías usando bases de datos relacionales

Al momento de hacer conversiones de datos relacionales a datos semánticos, diferentes problemas se presentan, entre ellos la posibilidad de definir de diferentes maneras las llaves primarias y foráneas, para la realización de conversión de datos relacionales a semánticos, diez diferentes combinaciones básicas de estas llaves deben ser consideradas al momento de realizar un mapeo directo, en la tabla 2 se muestra ésta combinación básica. De igual manera la evolución que ha sufrido el lenguaje SQL como puede ser visto en la tabla 2, ha incrementado la complejidad para procesar e interpretar de una manera simple la información almacenada en las bases de datos y convertirla a información semántica

3.1. Diferentes comandos utilizados por los diseñadores, producen diseños de bases de datos equivalentes, pero sintácticamente diferentes en el lenguaje SQL-DDL

Los elementos más importantes para analizar el esquema de las bases de datos son las llaves primarias y foráneas, estas llaves a su vez pueden ser simples o compuestas. Las diferentes combinaciones que se pueden presentar son mostradas en la siguiente tabla, Un análisis mas profundo debe ser realizado en cada caso a fin de convertir correctamente el esquema definido en información semántica.

Tabla 2. Consideraciones para producir datos semánticos usando el esquema relacional.

Combinación básica de llaves primarias y foráneas existentes en una base de datos relacional
PK: Una relación que contiene únicamente una llave primaria
C-PK: Una relación que contiene una sola llave primaria compuesta
S-FK: Una relación que contiene únicamente una llave foránea
N-FK: Una relación que contiene al menos dos llaves foráneas
PK + S-FK: Una relación que contiene una llave primaria y una llave foránea.
PK + N-FK: Una relación que tiene una llave primaria y dos llaves foráneas.
PK + N-FK: Una relación que tiene una llave primaria y más de dos llaves foráneas.
C-PK + S-FK: Una relación que tiene una llave primaria compuesta y solo una llave foránea
C-PK + N-FK: Una relación que tiene una llave primaria compuesta y dos llaves foráneas
C-PK + N-FK: Una relación que tiene una llave primaria compuesta y más de dos llaves foráneas

3.2. Diferentes comandos usados por diseñadores, para producir diseños de bases de datos equivalentes, pero sintácticamente diferentes en el SQL-DDL

Diversos autores están produciendo información semántica diferente basados en un mismo esquema de la BD, causando cierta incertidumbre para los usuarios, sin embargo actualmente los investigadores buscan un consenso para producir siempre todos el mismo resultado, en el trabajo realizado por Sequeda et al [3], son analizados los trabajos realizados por los autores más emblemáticos considerando principalmente la estructura básica de mapeo en un lenguaje semántico, las formas de interpretar atributos y restricciones, la implementación de la herencia en el lenguaje semántico (LS), la forma en que se interpreta la relación entre las tablas, el lenguaje que es utilizado para almacenar la información semántica generada, y otros aspectos de relevancia.

3.3. La evolución en el alcance del SQL ocasiona un diseño semántico más complejo

Desde la creación de la primer versión de SQL en 1986, el lenguaje ha evolucionado, permitiendo el soporte a bases de datos distribuidas, soporte de XML, manejo recursivo de *triggers* (disparadores), soporte de la tecnología orientada a objetos, además de incrementar la relación semántica entre los datos. Muchos de los analizadores de SQL-DDL no soportan las diferentes versiones existentes entre estos estándares, aún el SQL92 se considera un importante desafío para la automatización semántica, esto debido a la complejidad involucrada en estos comandos, en la actualidad no existe ningún desarrollo que pueda interpretar *triggers* en la tecnología semántica, esto principalmente debido a la complejidad involucrada en estos comandos. Adicionalmente cada productor de software de bases de datos implementa el

software con pequeñas diferencias, siendo necesario que sea desarrollada una versión por cada motor de base de datos desarrollado.

Tabla 3. Características más importantes de versiones de SQL estándar.

Año	Nombre	Alias	Comentarios
1986	SQL-86	SQL-87	Primera liberación realizada por ANSI.
1992	SQL-92	SQL2, FIPS 127-2	Revision Mayor (ISO 9075), SQL-92.
1999	SQL:1999	SQL3	Se agregan expresiones regulares. Triggers y consultas recursivas. Soporte de aspectos orientados a objetos.
2003	SQL:2003	SQL2003	Se introducen aspectos relacionados con XML Aspectos de secuencias estandarizadas y columnas con valores autogenerados.
2006	SQL:2006	SQL 2006	Son definidas diferentes formas en que SQL puede usarse en conjunción con XML. Importación, almacenamiento y manipulación de XML en una base de datos en SQL. Proporciona aplicaciones para integrar en código SQL el uso de XQuery, el lenguaje de consultas XML publicado por el W3C. Acceso concurrente a datos de SQL y documentos XML.
2008	SQL:2008	SQL 2008	Legaliza la instrucción ORDER BY fuera de los límites de cursores definidos. Incrementa la instrucción INSTEAD OF en triggers. Incrementa el estatuto TRUNCATE .
2012	SQL:2011		En proceso de liberación.

3.4. Lenguajes de ontologías más importantes que permiten expresar diferentes relaciones de datos usando diferentes comandos

Para clarificar las diferencias existentes entre dos de los lenguajes de ontologías más importantes (RDFS y OWL), es importante indicar el rol de cada lenguaje. Primeramente el RDF (Resource Description Framework) define una estructura adicional a las tripletas, las cuales básicamente se conforman de (Sujetos, Predicados y Objetos), el elemento de más alta relevancia en esta representación, es el predicado denominado "rdf:type". El cual se usa para describir que cada atributo de la base de datos, pertenece a un tipo específico de datos.

El lenguaje RDFS (RDF Schema) define clases que representan el concepto de los sujetos, objetos y predicados, esto permite comenzar a crear estatutos acerca de clases de cosas y tipos de una inter-relación. Usando estas clases y tipos es posible establecer relaciones entre dos clases, permite también describir en un lenguaje de texto entendible por humanos el significado de una relación o una clase. Permite igualmente describir los usos legales de varias clases e interrelaciones, sirve igualmente para indicar si una propiedad de clase es un subtipo de un tipo más general. OWL incrementa semántica al esquema, permite especificar más información acerca de las propiedades y las clases, algo importante es que permite expresar la información en tripletas, es posible indicar la transitividad. Permite igualmente el uso de sinónimos, manejo de cardinalidad, rango de datos y muchos comandos más para cono-

cer más específicamente las clases de comportamiento. En la figura 3 se muestra la estructura propuesta por el W3C, puede observarse la relación existente entre RDF, RDFS y OWL y que mientras más se sube de nivel en la figura, los datos producidos son más significativos.

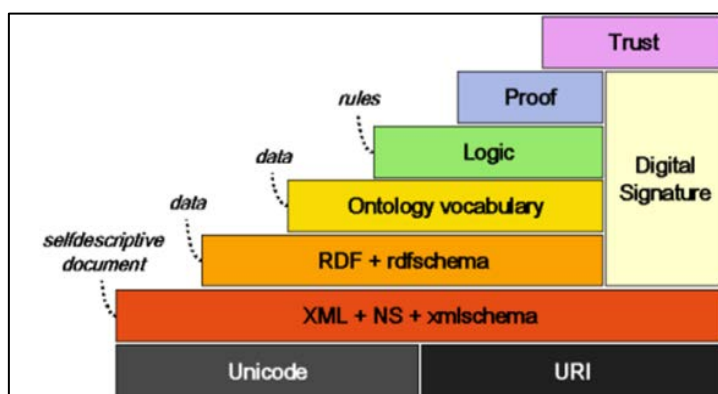


Fig. 3. Niveles en la web semántica propuestos por la W3C.

4. Formulación de ontologías

La definición mas popular de ontología es la propuesta por Gruber: “Una especificación formal explícita de una conceptualización compartida” [14]. *Formal* significa que la especificación está codificada en un lenguaje cuyas propiedades formales son bien entendidas; una *especificación explícita* significa que los conceptos y las relaciones en un modelo abstracto proporcionan nombres explícitos y definiciones. Una *Conceptualización* en este contexto, se refiere a un modelo abstracto de como las personas piensan acerca de las cosas, es usualmente restringido a un área particular. *Compartido* se refiere a que el propósito principal de una ontología es para ser usada y reutilizada entre diferentes aplicaciones y comunidades.

Diferentes razones hacen necesaria la creación de una ontología: a) para tener un entendimiento común de la estructura de la información entre personas y agentes de software; b) para habilitar el reuso del dominio de conocimiento; c) para hacer las afirmaciones del dominio explícitas; d) para separar el dominio del conocimiento del conocimiento operacional; e) para analizar el dominio de conocimiento.

Existen muchas discusiones acerca del significado exacto de una ontología, sin embargo existe una coincidencia en las siguientes dos descripciones: a) Es considerado como un vocabulario de términos que se refieren a puntos de interés en un dominio determinado; b) es una especificación acerca del significado de términos basados en algún tipo de lógica. Una ontología junto con algún conjunto de instancias concretas de una clase constituye una base de conocimientos. La taxonomía de clases es un punto fundamental en cada ontología. Es necesario enfatizar que la construcción de una ontología consiste de 6 pasos [20]: 1) Especificación del propósito, uso, alcance y grado de formalidad de la ontología; 2) Colección de datos usando varios métodos de recolección; 3) Conceptualización de términos del dominio (ontología preliminar); 4) Integración con otras ontologías; 5) Formalización en un lenguaje de ontologías; 6)

Evaluación de completitud, consistencia y redundancia. Estos pasos deben ser considerados como base para producir ontologías confiables para el propósito buscado.

4.1. Tipos de ontologías

Existen diferentes formas de clasificación de las ontologías, una posible clasificación se basa en la manera de especificar el significado de los términos. En la figura 4 se muestra en un extremo las ontologías poco formales adecuadas para tareas sencillas, en el extremo contrario se muestran las ontologías extremadamente rigurosas, formalizadas con teorías lógicas formales, estas ontologías proporcionan un soporte para el razonamiento automatizado siendo adecuadas para el descubrimiento de conocimiento.

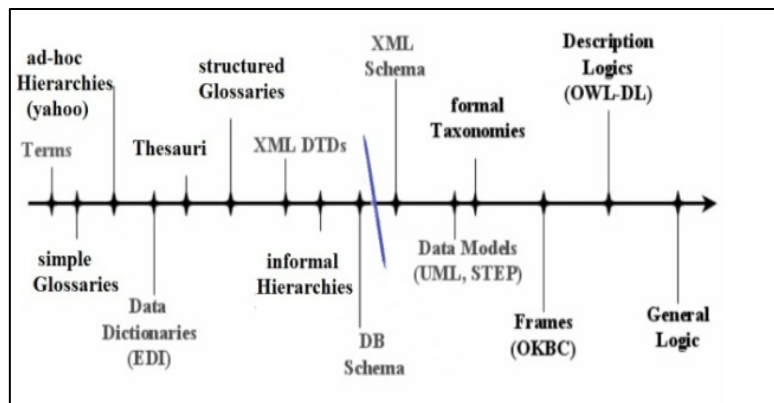


Fig. 4. Clasificación de Ontologías en base a su complejidad

Otra clasificación es con el uso de escenarios de aplicación, siendo de primordial importancia quien interactúa con la ontología producida, en este caso cuatro categorías pueden ser mencionadas de acuerdo a la clasificación realizada por Uschold and Gruninger [15]. Otras posibles clasificaciones de ontologías que pueden ser consideradas son las siguientes: a) De generales a específicas (Genéricas, de Nucleo, de Dominio, de Tareas, y de Aplicación); b) relacionadas con los lenguajes de representación; c) basadas en lógicas de descripción (DL), las cuales pueden ser usadas para representar terminología de conocimiento de una aplicación de dominio en una forma formal y estructurada; d) Caracterización de Ontologías DL en base al constructor utilizado. Las ontologías pueden utilizar varias de estas clasificaciones al mismo tiempo, el incremento de constructores en una ontología incrementa la complejidad de una ontología producida, es por esto que es necesario antes de crear una ontología un buen conocimiento del problema a resolver Kepler et al [16].

4.2. Ontologías y bases de datos

En vista de que los esquemas de las BD son documentos formales diseñados por especialistas en ingeniería de software que buscan modelar el mundo real, es posible producir una ontología a partir de éstos, las ontologías están formadas por una se-

cuencia de anotaciones, axiomas y hechos, los elementos más esenciales son los axiomas y los hechos, los cuales proporcionan información acerca de las clases, propiedades e individuos, de acuerdo a Kashyap [19], la construcción de ontologías a partir de BD relacionales requiere analizar el esquema y determinar Llaves primarias y foráneas y dependencias de inclusión. Se considera que el lenguaje OWL-LITE es suficientemente poderoso para representar la información almacenada en el esquema, aspectos tales como: Igualdad de términos, características de propiedad, restricciones de propiedad, restricciones de cardinalidad, información de encabezados, intersección de clases, control de versiones, propiedades de anotación y tipos de datos son elementos expresables en el lenguaje para crear una ontología de una base de datos de intranet.

5. Generación semiautomática de ontologías

Como se mencionó en la sección 2, la conversión automática de información contenida en las bases de datos relacionales, se encuentra todavía bajo un proceso de refinamiento, es por esto que la mayor parte de desarrolladores de ontologías utilizan la generación semiautomática de ontologías, proceso que puede conformarse por tres pasos para preparación de la información y al menos otro paso para la explotación de la información semántica producida, a continuación se mencionan los pasos requeridos para generar un ontología de manera semiautomática

Conversión de la estructura de bases de datos relacionales

Archivos RDF (conversión física de datos). Muchos productos de software de tecnología semántica, han adoptado este método, en el cual el total de los datos son convertidas a tripletas de información (s,p,o), una vez realizada la conversión, se incluye información acerca de la estructura de los datos, usando generalmente el lenguaje RDFS, y procediendo posteriormente a alinear la información producida a una ontología de dominio existente (wrapping systems), algunos ejemplos de este enfoque pueden encontrarse en : Arens et al 1993 [4], Barraza et al 2006 [5] y Zhou et al [6].

Archivos Map (Conversión de la estructura de datos). En este enfoque el esquema de SQL es transformado a una ontología sencilla, pudiéndose tener 2 enfoques diferentes, el primero sería que los datos se dejan en las bases de datos y se establecen mecanismos para mapear los datos relacionales a semánticos, y el segundo es que los datos relacionales son generados como instancias de la ontología creada, ejemplos del primer caso se pueden encontrar en D2RQ, y del segundo caso en los autores comparados en el trabajo realizado por Sequeda et al [3] y por autores como Shen et al.

Generación de ontologías

Generación manual de ontologías. El proceso de crear ontologías, es generalmente realizado utilizando editores de ontologías, los cuales permiten organizar gráficamente los conceptos componentes de la ontología, existen muchas herramientas para realizar ésta actividad, dentro de las principales podemos mencionar a Protegé, Swoop, Topbraid, etc. Estas herramientas nos ayudan a generar las partes repetitivas

de la ontología, evitando la generación de errores debido a fallas en la sintaxis, permitiendo visualizar la ontología en el momento en que se está creando.

Generación automática de ontologías. Este tópico requiere tener un completo dominio del estándar SQL-DDL, el cual permite generar la asociación existente entre los datos almacenados y la base de datos en términos semánticos, en este enfoque existen diversos desajustes, debido a que diferentes autores interpretan los comandos SQL de manera diferente, o utilizan diferentes comandos para generar las ontologías. Actualmente se busca la estandarización en esta área para producir exactamente las mismas ontologías

Interconexión de datos semánticos con ontologías

En la actualidad este aspecto es realizado manual o semiautomáticamente, produciendo una salida de los datos relacionales y creando una asociación con la ontología utilizando un editor de ontologías, ésta asociación requiere la consideración de similitudes, como se menciona en los trabajos de Buttler [8] y de Todorov y Geibel [9], donde se consideran los siguientes aspectos: a) Similitud en strings, b) Similitud en sinónimos, c) Similitud basada en instancias. Para la alineación de términos, se ha considerado la utilización de dos tipos de similitud (a y b) debido a que el tipo de aplicación a integrar, está más relacionada con la estructura de los datos y no tanto con el contenido de los atributos. En el trabajo propuesto se considera la utilización de 2 tipos de similitud, más relacionados con la estructura de los datos que con las instancias.

Resolución de consultas utilizando la tecnología semántica

Esta parte del trabajo, se encuentra actualmente en desarrollo, se están considerando dos diferentes opciones: a) Utilizar el lenguaje de consultas semánticas en archivos RDF (SPARQL), b) Desarrollar un lenguaje de consultas semánticas aplicado a bases de datos relacionales.

6. Areas de investigación futuras

Existen en la actualidad varios problemas abiertos, los cuales pueden categorizarse en tres grandes grupos, tales como:

- **Mapeo de Esquemas:** En este proceso, no obstante el gran número de productos existentes, se requiere realizar algunos refinamientos, debido a que no existen suficientes mecanismos para mapear la información de las bases de datos, tampoco existe soporte para todos los manejadores de BD, sin embargo existe soporte para la mayor parte de los productos más conocidos. El lenguaje para realizar ontologías más actual es el OWL, el cual conforma las ontologías con una secuencia de anotaciones, axiomas y hechos. Los elementos más relevantes son los axiomas y los hechos, ya que proporcionan información acerca de las clases, propiedades e individuos, importantes aspectos a considerar para producir una ontología exitosa, son detallados por Du Bois et al [20].

- **Razonamiento con búsquedas imprecisas:** Una de las fuerzas de la utilización de ontologías, es la relacionada con la asociación de términos en una jerarquía dentro de un dominio de conocimiento, esto permite el uso de sinónimos y búsquedas más generalizadas, proporcionando una base sólida para el conocimiento estructurado, con la infraestructura semántica es posible manejar términos equivalentes, así mismo permite realizar actividades de agregación o desagregación como se necesite [22].
- **Consultas Semánticas:** Una comparación entre los lenguajes de consultas de las BD y los lenguajes de consulta de ontologías, es necesaria para seleccionar la mejor estrategia para la realización de consultas semánticas. Dos diferentes estrategias pueden ser utilizadas en los lenguajes de consulta ontológicos: 1) Uso exclusivo de clases y propiedades, los cuales se utilizan para capturar relaciones intencionales, siendo pobres como lenguajes de consultas, debido a que no es posible referirse al mismo objeto por diferentes trayectorias de navegación dentro de la ontología. 2) Full SQL (equivalente a lógica de primer orden), tiene el inconveniente de ser indecidible cuando se tiene información incompleta Giacomo et al [23], proponen el uso de uniones de consultas conjuntivas (CQs).

7. Conclusiones

La extracción de información almacenada en bases de datos usando ingeniería inversa para su conversión a información semántica, requiere que la BD esté en al menos en 3NF (tercer forma normal), debido a que esto ayuda a asegurar que el diseño está adecuadamente realizado, este proceso involucra que una vez que se tiene toda la información automatizada, se cuenta con un uso adecuado de llaves primarias y una correcta asociación entre tablas. Sin embargo un procesamiento adicional debe ser hecho para contar con una ontología funcional, es necesario trabajar con los pasos recomendados por Du Bois [20] para crear las ontologías, lo cual no se considera una tarea sencilla. La elaboración de ontologías requiere parsear el script de SQL y comenzar identificando los elementos relevantes necesarios para crear la ontología. El grupo incubador W3C [12] continua la investigación a través del grupo RDB2RDF para mejorar el proceso de conversión automática de datos, la integración de información de diversos repositorios, optimización de almacenamiento y optimización de consultas. Es importante mencionar que de las dos alternativas para producir ontologías, consideramos el “mapeo directo” como la mejor alternativa a ser realizada, aunque no es el más popular debido a su complejidad, es importante mencionar que la generación automática aún se encuentra bajo investigación y desarrollo [10], una estandarización es fundamental en el área, la cual todavía no se ha consolidado, aun cuando las ontologías sean producidas manualmente, estas no están totalmente aceptadas por la comunidad de productores de ontologías, como puede ser visto en la investigación de Todorov and Geibel [9]. La creación automática de ontologías no se encuentra todavía estandarizada, existen diferentes iniciativas para mejorar el soporte y desempeño de consultas cuando existe información incompleta, una de las características para el logro de la integración semántica es el incremento de posibilidades para la explotación de consultas así como una simplificación en el proceso de explotación.

Referencias

1. Jung An, Geller J, Wu Y, Choon S.: Semantic Deep Web: Automatic Attribute Extraction from the Deep Web, Data Sources, ACM 1-59593-480 (2007)
2. Berners Lee, Tim; Fischetti, Mark: Weaving the web. Harper San Francisco, chapter12, ISBN 978-0-06-251587-2 (1999)
3. Sequeda J., Tirmizi S-, Corcho O., Miranker D.: Survey of directly Mapping SQL databases to the Semantic Web. The Knowledge Engineering Review, Cambridge University Press, Vo 26:4, 445-486 (2011)
4. Arens, A. Chee, C. Y., Hsu, C & Knoblock, C.: Retrieving and integrating data from multiple information sources. International Journal on Intelligent and Cooperative Information Systems, 127-158, (1993)
5. Barrasa, J., Gómez Pérez, A.: Upgrading relational legacy data to the semantic web. In: proc. of 15th international conference on World Wide Web Conference (WWW 2006), pages 1069,1070, Edinburgh, United Kingdom (2006)
6. Zhou Shufeng: Mapping Relation Databases for Semantic Web. International Conference on Future BioMedical Information Engineering (2009)
7. Shen, G., Huang, Z., Zhu, X. & Zhao, X: Research on the rules of mapping from relational model to OWL. In: Proceedings of the workshop on OWL: Experiences and Directions, Athens, GA, USA (2006)
8. D. Buttler: A short Survey of Document Structure Similarity Algorithms. International Conference on Internet Computing, Las Vegas, NV, United States (2004)
9. K. Todorov, P Geibel: Ontology Mapping via structural and Instance-Based Similarity Measures. Third International Workshop On Ontology matching, OM2008 (2008)
10. D O'Leary: Different Firms, Different Ontologies, and No one Best Ontology. IEEE Intelligent Systems, IEEE, pp 72-78 (2000)
11. W3C Semantic Web Activity, <http://www.w3.org/2001/sw/>
12. W3C RDB2RDF Incubator Group 2009, http://www.w3.org/2005/Incubator/rdb2rdf/RDB2RDF_SurveyReport.pdf
13. A. Doan and A. Halevy: Semantic-Integration Research in the Database Community. AI Magazine Volume 26 Number 1 (2005)
14. Gruber, T.: A translation approach to portable ontology specifications. Knowledge Acquisition 5:199-220 (1993)
15. M. Uschold, M. Gruninger: Ontologies and Semantics for Seamless Connectivity, SIGMOD Record, Vol. 33, No. 4 (2004)
16. F. Kepler, C. Paz-Trillo, J. Riani, M. Moretto , K. Valdivia-Delgado, L. Nunes, and R. Wassermann: Classifying ontologies. In Proceedings of the Second Workshop on Ontologies and their Applications (WONTO 2006) (2006)
17. OWL Web Ontology Language Overview, W3C Recommendation 10 February (2004)
18. Asunción Gómez-Pérez and Oscar Corcho: Ontology Languages for the Semantic Web. IEEE INTELLIGENT SYSTEMS (2002)
19. Kashyap, V.: Design and creation of ontologies for environmental information retrieval. 12th Workshop on Knowledge Acquisition Modeling and Management (KAW'99). Banff, Canada, October 1999, <http://sern.ualgary.ca/ksi/kaw/KAW99/papers/Kashyap1/kashyap.pdf> (2007-03-15).
20. Du Bois, B.: Towards an ontology of factors influencing reverse engineering. In: STEP '05: Proceedings of the 13th IEEE International Workshop on Software Technology and Engineering Practice, USA, pp. 74-80 (2005)
21. Bizer C., Cyganiak, R.: D2RQ — Lessons Learned. Position paper for the W3C Workshop on RDF Access to Relational Databases, Cambridge, USA (2007)

22. Davidson, James y Jerrold Kaplan: Natural Language Access to Data Bases: Interpreting Update Requests. Computer Science Department, Stanford University, California, ACM Press (1983)
23. G de Giacomo, Towards Systems for Ontology-based Data Access and Integration using Relational Technology, Sapienza Universita di Roma, U of Toronto (2010)

Sistema basado en conocimiento para la industria del vestido en México

Karina Santiago-Santiago¹, Ana Lilia Laureano Cruces², Jorge Sánchez de Antuñano²,
Estela Sarmiento Bustos¹ y Oscar Domínguez-Pérez¹

¹ División Académica de Mecánica Industrial, Universidad Tecnológica Emiliano Zapata, Av.
Universidad No.1 Col. Palo Escrito, Emiliano Zapata, Morelos, 62790, México
karysanty@hotmail.com

² Área de Posgrado en Diseño, Edificio D 1er. Piso, Universidad Autónoma Metropolitana,
Unidad Azcapotzalco, Av. San Pablo 180 Col. Reynosa Tamaulipas, 02200, México DF,
México

Resumen Hoy en día la Industria del Vestido en México es susceptible a problemas complejos. Este tipo de industria está sujeta a las influencias que con el paso del tiempo modifican la percepción de los involucrados en el proceso de diseño, dado que nos enfrentamos a problemas con características objetivas y subjetivas. En este trabajo se utilizaron las técnicas de entrevista, método de observación directa y argumentos teóricos para obtener el conocimiento del experto en el dominio, el cual define los problemas que surgen en el proceso de diseño de prendas de vestir para mercados mexicanos. El presente trabajo tiene como objetivo generar una metodología en donde el experto del dominio será un resultado altamente especializado al sustraer sus habilidades y reforzarlas con la Metodología de Sistemas Suaves y Modelo de Gestión de Diseño. Los resultados sugieren que es aplicable a cualquier campo del diseño.

Palabras clave: Sistema, conocimiento, industria, sistémico, sistemático.

1. Introducción

En México, la Industria del Vestido ha desempeñado un papel destacado dentro de la sociedad y la economía debido a su importante participación económica en el producto interno bruto (PIB) y en el empleo nacional [1]. Actualmente esta industria funciona como un sistema complejo, un problema genera otro problema, las influencias a las que están expuestas con el paso del tiempo modifican su percepción. Su complejidad se deriva de los problemas duros como de los problemas suaves entre éstos: débil integración de los procesos productivos en bienes de alto valor agregado, baja capacidad para desarrollar productos diferenciados con moda y calidad, debilidad para evolucionar de la maquila de productos básicos a productos competitivos, resistencia al cambio, bajo nivel educativo y cultural del personal así como deficiencia en el uso de las nuevas tecnologías.

La ingeniería de software (enfoque sistemático) ha contribuido en el funcionamiento del sistema de la industria del vestido, estos van desde sistemas de

información, programas de diseño en 2D y 3D, sistemas CAD-CAM hasta programas especializados de dibujo, trazo y patronaje (Lectra Systemes, Gerber Technology, Audaces, Optitex). Sin embargo, desde la ingeniería del conocimiento (enfoque sistémico) no existen programas que contribuyan a su funcionamiento [1].

Los expertos del dominio en la industria del vestido suelen estar relacionados con la práctica del campo más que con el sustento teórico, la información está estructurada de un modo cognitivamente complejo por lo que resulta difícil extraer el conocimiento y esto representa un cuello de botella para la construcción de un Sistema Basado en Conocimiento (SBC). La adquisición del conocimiento de manera incompleta influye negativamente en la calidad del SBC, de lo contrario el éxito de la adquisición del conocimiento depende en parte de la capacidad de conceptualizar correctamente el dominio de la aplicación consecuentemente asegura la calidad del SBC.

Con base en lo anterior, el objetivo del presente trabajo es proponer una metodología para extraer el conocimiento del experto en el dominio en la búsqueda, identificación y descripción de problemas complejos y propuestas de planes de mejora para optimizar el proceso de diseño de prendas de vestir en México mediante las técnicas de entrevista, observación directa, cuestionarios y argumentos teóricos [2, 3].

2. Teoría

El éxito o fracaso del SBC depende directamente de la adquisición del conocimiento, para ello existen tres grandes grupos de técnicas que se utilizan para extraer el conocimiento: manuales, semiautomáticas y automáticas. Dentro de las técnicas manuales destacan: las entrevistas, análisis de protocolos, cuestionarios, la observación directa y la extracción de curvas cerradas. Dependiendo del tipo de conocimiento debe elegirse la técnica para poder extraerlo con éxito.

El conocimiento del experto del dominio en el proceso de diseño en la industria del vestido en México es un conocimiento de tipo procesal y al mismo tiempo episódico, es decir, en parte el procedimiento está muy automatizado, un paso del proceso dispara el siguiente y al mismo tiempo está basado en la experiencia y es de carácter autobiográfico.

La técnica de entrevista es el método de adquisición más usado, ya que permite adquirir conocimiento en cualquier campo, diferentes tipos de conocimientos y a distintos niveles del proceso de desarrollo del SBC [4]. En cuanto su orden temporal y su contenido las entrevistas se clasifican en: iniciales, no estructuradas y estructuradas.

La técnica de observación directa consiste en observar al experto mientras trabaja en sus tareas habituales, en situaciones reales. En la observación directa no se realizan interpretaciones subjetivas, la técnica es útil cuando la actividad del experto se basa en seguir un guion o protocolo y carece de tal utilidad cuando el trabajo del experto es tipo conceptual [4, 5].

Los cuestionarios son una técnica que consiste en presentarle al experto una serie de fichas en donde se le plantean preguntas muy concretas que éste debe resolver. Los

cuestionarios permiten que el experto pueda asesorar sus respuestas con consulta de libros, revistas, etc. [5, 6].

Debido a que el experto del dominio carece de ciertos conocimientos y habilidades para la búsqueda, definición y solución de problemas complejos en la industria del vestido en México la información faltante se extrae a través de la Metodología de Sistemas Suaves y el Modelo de Gestión del Diseño [2, 3].

El modelo fue propuesto por Checkland [2] a partir de una adaptación de un Modelo de la Innovación [3], éste determina el diagnóstico del proceso de gestión del diseño y propone planes de mejora para optimizarlo, consecuentemente los resultados aseguran el éxito económico. El método de investigación consiste en una serie de entrevistas a profundidad aplicadas a los directivos que intervienen en el proceso de diseño, el análisis de los datos secundarios (manuales, página web e informes técnicos) y la observación no intrusiva. El modelo de gestión del diseño lo engloban cinco actividades: Cultura corporativa y orientación al diseño, generación de conceptos, estrategia de diseño, recursos e implementación y resultados, que se presentan en la Fig. 1.



Fig. 1. Modelo de Gestión del Diseño.

La Metodología de Sistemas Suaves (Soft System Methodology) o Metodología Sistémica aborda problemas suaves y propone cambios sobre el sistema para mejorar su funcionamiento, es una manera muy útil de acercarse a situaciones complejas sociales, y encontrar las respuestas a sus problemas correspondientes. La MSS fue desarrollada por Checkland, al comprender que la sistemática no era apropiada para investigar a los sistemas organizacionales poco estructurados, grandes y complejos. No señala en sí misma las soluciones, o los criterios para llegar a la solución óptima, sino comienza investigando los hechos de la situación del problema hasta llegar a las posibles soluciones que mejorarán el funcionamiento del sistema. Este proceso se desarrolla en siete etapas, divididas en dos actividades: las etapas 1, 2, 5, 6 y 7 que corresponden a las “actividades del mundo real”; y las etapas 3 y 4 que son actividades del “pensamiento de sistemas”, como se muestra en la Fig. 2.



Fig.2. Modelo de Sistemas Suaves.

En la Tabla 1 se muestran las ventajas y desventajas que presentan éstas técnicas.

Tabla 1. Ventajas y desventajas de las técnicas para adquirir conocimiento.

Técnica	Ventaja	Desventaja
Entrevista	Se adquiere gran cantidad de conocimiento.	Consume gran cantidad de tiempo, se basa en la introspección.
Observación directa	No es introspectiva, se obtiene la información directa del trabajo del experto.	Consume mucho tiempo es largo y costoso.
Cuestionarios	Resulta poco costosa y consume poco tiempo.	Es de tipo introspectiva, se requiere estar familiarizado con el dominio para elaborar las preguntas.
Argumentos teóricos (MSS y MGD)	Se basa en casos de éxito.	Se requiere estar familiarizado con el dominio.

3. Método propuesto

Para adquirir el conocimiento, se tomó un caso de estudio de la base de datos de la Cámara Nacional de la Industria del Vestido Delegación Morelos, las características de la empresa proporcionada son las siguientes:

- Empresa fabricante de prendas de vestir.
- Cuenta con el departamento de diseño.
- Tiene más de 7 años de antigüedad.

Fase I. Diseño exploratorio

Diseño de los protocolos para la recolección de datos, recopilación de información y análisis inicial de un caso de estudio en el sector de la industria del vestido. Desarrollo del estudio de caso (contacto con la empresa y expertos del dominio).

Fase II. Obtención de la información: cuestionario, entrevista y observación directa

Utilizando la técnica de cuestionario se plantean las preguntas muy concretas que el experto del dominio deberá resolver y que serán utilizadas en la entrevista. Para ello, se construyen preguntas abiertas y preguntas cerradas. Se acude a las instalaciones donde labora el experto del dominio para captar el conocimiento procesal sin perturbar las condiciones de trabajo normales del experto, se toman notas de las observaciones y posteriormente se inicia con las entrevistas.

La entrevista se desarrolla en tres fases, la primera es una entrevista inicial que servirá para establecer una buena relación con el experto en el dominio. Esta es una entrevista de uno a uno, aproximadamente de una hora que servirá para contextualizar el sistema. Posteriormente, se inicia en otra sesión con una entrevista no estructurada a base de preguntas abiertas, la duración de la entrevista es aproximadamente de 1 y 3 hrs. Una vez realizadas las entrevistas anteriores se procede a iniciar con la entrevista estructurada en la que se pretende adquirir y entender el conocimiento específico utilizado por el experto para solucionar los problemas del campo.

Fase III. Búsqueda de las fuentes de información: Metodología de Sistemas Suaves (MSS) y Modelo de Gestión del Diseño (MGD)

Por medio de la MSS y el MGD se extrae el conocimiento episódico basado en hechos y situaciones en treinta y una empresas españolas de tres sectores: mobiliario, electrónica y turismo. El MGD se adapta al sector de la industria del vestido en México, y contribuye a desarrollar la Etapa 1, 2, 3 y 4 de la MSS al que le denominamos Exitus, como se muestra en la Fig. 3.

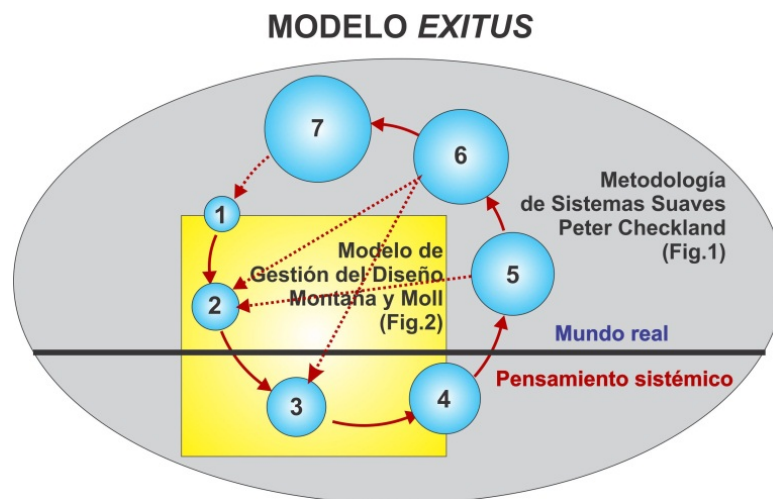


Fig. 3. Modelo Exitus.

Fase IV. Análisis de los datos

Transcripción de las entrevistas y análisis del material. Análisis transversal de la información (entrevistas, cuestionarios, observación directa y fuentes de información). La información se analiza y se segmenta en trozos o frases cortas que tengan sentido independientemente. Posteriormente, se identifican los objetos, sus valores y las relaciones entre los mismos. Se construyen las reglas en función de la base de conocimiento.

4. Resultados

4.1. Base de conocimiento

La base de conocimiento tiene una estructura de reglas, clases y atributos. El primer bloque de reglas corresponde a la selección de diagnóstico empresarial, éste es un módulo general, y comprende las fases de: cultura, concepto, registro, estrategia, recursos y resultados (MGD).

Tabla 2. Clases y atributos en la construcción de reglas.

Clase: Fase de Recursos	
Atributo	Valor de atributo
Aspecto	Trabajo, presupuesto, conocimiento, equipo
Trabajo en equipo	1,2,3,4
Responsable	1,2,3,4
Conocimiento	1,2,3,4
Formación	1,2,3,4
Diagnóstico	Nunca, ocasionalmente, a menudo, siempre
Propuestas de solución	Fomentar la comunicación, crear equipo de diseño interno, existe comunicación del equipo interno con el externo. Designar un responsable, formalizar un departamento de diseño, fijar presupuesto. Registrar conocimiento, sistematizar el conocimiento, registrar los diseños Apoyar actividades de formación, fijar presupuesto de formación, establecer programa de carrera

Cada fase la componen cuatro aspectos, que en su conjunto definen la totalidad de cada una de las fases y comprenden el segundo bloque que sirve para buscar el problema (MSS). Posteriormente, se expandirá la base de conocimiento hacia otros tipos de aspectos que contribuyan a la búsqueda de problemas complejos. El tercer bloque tiene como función describir el problema como resultado del diagnóstico y el cuarto bloque consiste en proponer las alternativas de solución o planes de acción.

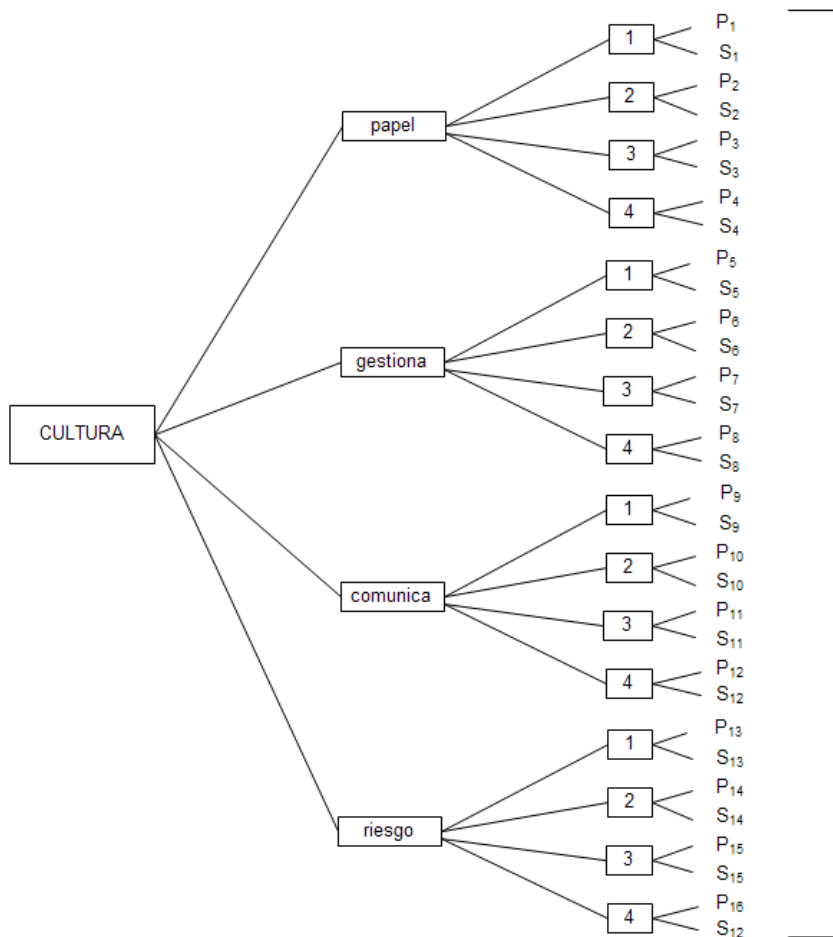


Fig. 4. Fragmento del árbol de CULTURA para definir el problema y proponer una solución.

Como un ejemplo en la Tabla 2 se muestran la clase de Fase de Recursos y sus atributos para construir las reglas.

Con base a lo anterior, se mencionan las siguientes reglas:

Regla 1: Si formación es 2, entonces la formación del equipo de diseño ocasionalmente ha sido considerada.

Regla 2: Si conocimiento es nunca, entonces se requiere registrar el conocimiento.

Regla 3: Si responsable es 2, entonces ocasionalmente ha existido un presupuesto de diseño y un responsable de diseño.

Regla 4: Si conocimiento es 3, entonces a menudo ha sido documentado, la información está disponible en la empresa.

Regla 5: Si recursos es a menudo, entonces se requiere fortalecer las actividades de formación, dar seguimiento a programas de carrera e incrementar el presupuesto de formación.

Por otra parte, en la figura 5 se presenta un fragmento de árbol de la búsqueda y descripción del problema (P1, P2,... Pn), así como las propuestas de solución (S1, S2,... Sn).

4.2. Propuesta de evaluación de la base de conocimiento

Con base en un ejercicio de un caso de estudio práctico, se recaba la información, y se generan dos documentos para su análisis, uno realizado por el experto en el dominio y otro a partir de las reglas extraídas de la Base de Conocimiento. Ambos resultados se comparan para verificar la efectividad del sistema. Como se muestra en el siguiente ejemplo:

Problema complejo de la empresa. ¿Por qué no se utilizan en su totalidad los sistemas CAD en el proceso de diseño de prendas de vestir?

Siguiendo el Modelo Exitus, se buscan las situaciones o hechos que se ven involucrados en el problema por medio de la herramienta de diagnóstico (MGD). Como se observa en la Tabla 3.

Tabla 3. Herramienta de diagnóstico.

FASES	ASPECTOS
Cultura	¿Qué papel juega el diseño en la planificación estratégica? ¿Cómo se gestiona el diseño el diseño? ¿Se gestiona el riesgo? ¿Cómo se hace?
Concepto	¿Qué fuentes de información y de creación de ideas se utilizan? ¿Se definen los cor ¿Cómo se hace? ¿Cuál es la relación entre diseño y marketing? ¿Qué relación existe funciones de la empresa?
Estrategia	¿En la estrategia de la empresa se contempla el diseño? ¿Existe estr productos?, ¿está definida? ¿Existe una estrategia integral de diseño? diseño, está planificado?
Recursos	¿Existe un equipo de diseño?, ¿cómo es interno/externo y cómo tral responsable y un presupuesto destinado al diseño? ¿Cómo se crean los conocimientos de diseño? ¿Se forma al equipo de diseño?, ¿ex para ello?
Resultados	¿Cuál es el grado de novedad de los proyectos de diseño? ¿Son los rest consistentes con el resto de productos y con la imagen de la empresa? evaluar el diseño? ¿Cuál es el resultado final de los nuevos diseños?

Dada las características de la pregunta en el ejemplo expuesto, observamos que es un aspecto a evaluar en la Fase de Recursos, sin embargo, esto no implica que las demás etapas estén aisladas ya que es importante contextualizar cada una de ellas.

Las respuestas a las preguntas realizadas en la Tabla 3 se registran con valor numérico de 1 a 4, donde el valor 4 = verdad, 3 = no todo es verdad, 2 = no todo es falso, 1 = falso. Como se muestra en la Tabla 4, a cada una de las respuestas se le

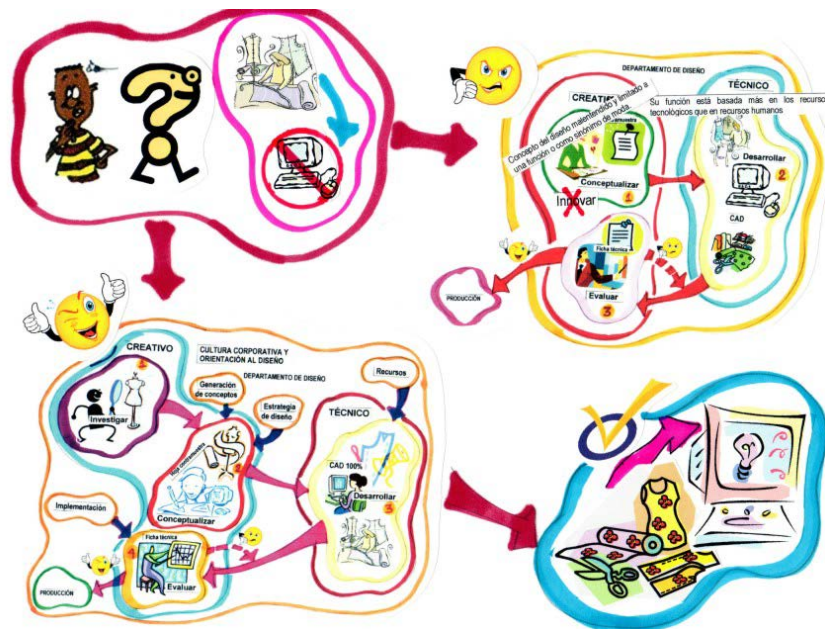


Fig. 5. Sistema de actividad humana.

asigna un valor, posteriormente se obtiene el valor promedio (se suman los valores asignados a las respuestas y se divide entre el número de respuestas) el cual determina el tipo de problema.

Tabla 4. Asignación de valores.

FASES	ASPECTOS
Cultura	¿Qué papel juega el diseño en la planificación estratégica? ¿Cómo se gestiona el diseño? ¿Cómo se comunica el diseño? ¿Se gestiona el riesgo? ¿Cómo se hace?
Concepto	¿Qué fuentes de información y de creación de ideas se utilizan? ¿Se definen los conceptos de producto?, ¿Cómo se hace? ¿Cuál es la relación entre diseño y marketing? ¿Qué relación existe entre diseño y otras funciones de la empresa?
Estrategia	¿En la estrategia de la empresa se contempla el diseño? ¿Existe estrategia de nuevos productos?, ¿está definida? ¿Existe una estrategia integral de diseño? ¿El proceso de diseño, está planificado?
Recursos	¿Existe un equipo de diseño?, ¿cómo es interno/externo y cómo trabaja? ¿Existe un responsable y un presupuesto destinado al diseño? ¿Cómo se crean y se transmiten los conocimientos de diseño? ¿Se forma al equipo de diseño?, ¿existe presupuesto para ello?
Resultados	¿Cuál es el grado de novedad de los proyectos de diseño? ¿Son los resultados del diseño consistentes con el resto de productos y con la imagen de la empresa? ¿Cómo se podría evaluar el diseño? ¿Cuál es el resultado final de los nuevos diseños?

Después de esto los datos son analizados a partir de dos criterios, la del experto y la base de conocimiento.

4.2.1. Análisis realizado por el experto en el dominio

Se construye un cuadro pictográfico (acopio y clasificación de la información) que representa el problema no estructurado y el sistema de actividad humana, como se observa en la Fig. 5.

Posteriormente se define la situación del problema no estructurado como se observa en la Tabla 5.

Tabla 5. Definición de la situación del problema no estructurado.

FASES	Situación del Problema
Cultura	La dirección no ve la necesidad de gestionar el diseño y lo ve como un coste adicional. La dirección habla de diseño, pero no consigue transmitir de forma coherente su visión del mismo
Concepto	Se trabaja sobre lo que ya se está haciendo y, en todo caso, se reacciona ante demandas concretas cuando aparecen. Marketing y diseño apenas tienen contacto. Poco contacto de diseño con otros departamentos
Estrategia	El diseño es importante para la empresa, pero no forma parte de la estrategia, no es una competencia básica. La empresa tiene una estrategia de nuevos productos poco definida, a menudo imitando a los competidores.
Recurso	Hay un responsable de diseño sin asignación presupuestaria. Pocos recursos. El conocimiento es implícito a cada miembro y no está documentado. No se registra nunca ningún diseño. No se contempla la formación dentro del equipo de diseño.
Implementación	No se busca ninguna novedad, sólo mantenerse en el mercado. Los resultados no son muy innovadores, pero consistentes con la imagen de la empresa.

Con la situación del problema no estructurado (ver Tabla 5) y la experiencia en el proceso de diseño de la Industria del Vestido éstas se integran y se construye una sola definición para responder a la pregunta inicial. Como se muestra a continuación:

Los programas de diseño asistido por computadora (CAD) no se manejan en su totalidad debido a que el departamento de diseño no contempla un programa de formación. Su función está basada más en los recursos tecnológicos que en los recursos humanos, el personal que opera los sistemas carecen de habilidades y tiempo para practicar; los conocimientos son implícitos en cada empleado y no están documentados. El concepto de diseño está mal definido y limitado a una sola función, se maneja como sinónimo de moda. Se habla de diseño pero no se logra transmitir la visión. La mayoría de las personas que intervienen en el proceso, consideran que diseñar es copiar prendas de marcas reconocidas, así el diseño de sus prendas y

patrones resultan básicos, como consecuencia no se utilizan todas las funciones que ofrece el sistema CAD, ver Fig. 5.

Posteriormente, por medio de la herramienta CATWOE (MSS) se construye la definición raíz equivalente a las propuestas de solución para expresar la función central del sistema de actividad humana, donde se encuentra inmersa la situación del problema como se muestra en la Tabla 6.

Tabla 6. Definición raíz del sistema de actividad humana con problema estructurado.

FASES	Propuestas de solución
Cultura	La dirección de la empresa se involucra en la gestión del diseño para innovar productos de calidad por medio de estrategias de diseño, administración adecuada de los recursos y un plan de evaluación de los resultados.
Concepto	Diseño, marketing y otros departamentos generan ideas para definir el producto en base a la captación de oportunidades y a las capacidades internas.
Estrategia	Diseño contribuye a los objetivos de la empresa, en base a la estrategia de desarrollo de nuevos productos, la estrategia de la marca.
Recurso	Diseño gestiona los recursos para la formación de los diseñadores, la creación de conocimiento y la protección del conocimiento en base a las necesidades del departamento.
Implementación	La dirección evalúa el diseño para verificar el grado de innovación del producto y el proceso de producción.

La definición raíz que corresponde a la pregunta planteada inicialmente, sugiere que: El personal de marketing, el diseñador de moda, el diseñador gráfico y el técnico patronista de la empresa, manejarán en su totalidad los sistemas CAD para transformar la mayor parte del proceso de diseño a mano al proceso de diseño asistido por computadora en base al diseño y su gestión, de tal manera que las respuestas a las exigencias de las tiendas comerciales sean las solicitadas (Fig. 5).

4.2.2 Análisis realizado a partir de la Base de conocimiento.

En base a la Tabla 2 se generan las siguientes reglas:

- Si trabajo en equipo es 4, entonces siempre existe un responsable.
- Si recursos es 2, entonces ocasionalmente ha existido un presupuesto de diseño.
- Si conocimiento es 1, entonces nunca ha sido documentado, la empresa depende del conocimiento del personal.
- Si formación es 1, entonces nunca ha sido considerada.

Finalmente: Si recursos es ocasionalmente, entonces diseño gestiona los recursos para la formación de los diseñadores, la creación de conocimiento y la protección del conocimiento en base a las necesidades del departamento.

De este modo podemos verificar que tan asertivas son las reglas respecto del razonamiento del experto en el dominio y al mismo tiempo nos permite valorar los atributos y sus ponderaciones así como la visión clara de dónde deberá crecer la base de conocimiento.

5. Conclusiones

A través de la metodología mostrada en este trabajo es posible adquirir el conocimiento del dominio en problemas complejos como en la industria del vestido en México y analizar estos bajo el enfoque de sistemas. Esto es aplicable a otras áreas del diseño.

Referencias

1. Patlán, Delgado y Abdel.: La industria del vestido en México.: Diagnóstico, prospectiva y estrategia. Centro de Estudios de Competitividad del Instituto Tecnológico Autónomo de México (CEC-ITAM), 54 (2008)
2. Checkland, P.: Pensamiento de sistemas, práctica de sistemas. México, D.F. Limusa, 125-132 (2006)
3. Montaña, J., Moll, I.: Éxito empresarial y diseño. Madrid, España. FEEPD. (2008)
4. A.Alonso, B. Guijardo, A. Lozano, J. Tomás, M. Taboada.: Ingeniería del Conocimiento. Aspectos Metodológicos. Pearson Educación, S.A, 343-387 (2004). Palgrave MacMillan. (2005)
5. G. Pajares, M. Santos.: Inteligencia artificial e ingeniería del conocimiento. Alfa Omega Grupo Editor S.A. de C.V. 155-157 (2006)
6. J. Palma, R. Marín.: Inteligencia Artificial. Técnicas, métodos y aplicaciones. McGraw-Hill, 803-810 (2008)

Agentes

Tres modelos computacionales de filosofía política

José Martín Castro-Manzano

Escuela de Filosofía,
Universidad Popular Autónoma del Estado de Puebla,
21 sur 1103 Barrio de Santiago, Puebla, 72410, México
josemartin.castro@upaep.mx

Resumen Presentamos los primeros avances de una serie de estudios de filosofía política comparada usando métodos computacionales. Mostramos una tipificación de tres modelos clásicos de filosofía política y también un conjunto de experimentos que ofrecen evidencias para responder un problema fundamental de filosofía política: qué teoría clásica es más plausible para explicar el origen de nuestras estructuras políticas. En particular nos centramos en la relación ganancia-compromiso.

Palabras clave: Agentes, simulación social, filosofía política.

1. Introducción

El futuro de la filosofía está junto al laboratorio. La filosofía del futuro es, entre otras cosas, una filosofía experimental. Esta manera relativamente novedosa de hacer filosofía está estrechamente relacionada, por sus métodos y antecedentes, con la filosofía exacta [6], por lo que encontrar una relación orgánica entre la filosofía y las ciencias de la computación es muy natural, especialmente por medio de la lógica y los métodos formales. De esta relación no sólo obtenemos hipótesis interesantes, teorías complejas y hasta aproximaciones formales que soportan nuestro entendimiento de la conducta inteligente –y racional–, también adquirimos las ventajas de las herramientas y las técnicas de la inteligencia artificial, la cual es usada para construir nuevas tecnologías pero también está pensada, desde ya, para modelar fenómenos cognitivos [10], y en particular, para representar conductas político-sociales que también constituyen un aspecto de la inteligencia.

Bajo este panorama es posible identificar, sin gran dificultad, diferentes hipótesis de filosofía política, métodos basados en agentes de simulación social y especificaciones formales para agencia racional. De manera más específica, la filosofía política ofrece los fundamentos de las diferentes hipótesis y teorías que han llenado nuestros esquemas culturales con ideologías, argumentos y tesis sobre nuestra conducta política [14]. En segundo lugar, la simulación se ha concentrado, tradicionalmente, en la simulación de eventos físicos y numéricos; sin embargo, hay una tendencia creciente en el área de la simulación social que

provee un método útil y comprensible para explorar, y en algunos casos hasta visualizar, fenómenos sociales que en principio pueden ser analíticamente derivables pero que no son directamente observables [2]. Y, finalmente, la conducta inteligente parece tener una explicación lo suficientemente apropiada en un marco teórico que da a las creencias, deseos e intenciones un lugar fundamental. Esta postura intencional [9] para modelar fenómenos cognitivos nos da niveles de explicación, predicción y coordinación que otros marcos teóricos no proporcionan naturalmente [5].

Dados estos antecedentes, en este trabajo presentamos los primeros avances de una serie de estudios sobre filosofía política comparada. Presentamos una tipificación de tres modelos clásicos de filosofía política y también un conjunto de experimentos que ofrecen evidencias para responder un problema fundamental de filosofía política: qué teoría clásica es más plausible para explicar el origen de nuestras estructuras políticas. En particular nos centramos en la relación ganancia-compromiso social. Nuestra contribución consiste, principalmente, en mostrar que la reciprocidad, a largo plazo, tiene mejores resultados que el egoísmo o el altruismo en términos de, como dirían los anglosajones, un *trade-off* entre ganancias y compromiso social a la vez que sugerimos una explicación de las causas de este balance.

El artículo está organizado de la siguiente manera. En la Sección 2 discutimos el papel de la filosofía política y cómo podemos usar la simulación social para tratar problemas de esta clase. En la Sección 3 describimos los modelos utilizados y en la Sección 4 mostramos los avances obtenidos hasta el momento. Finalmente en la Sección 5 mencionamos la dirección de investigaciones futuras.

2. Preliminares: filosofía política y simulación

2.1. Filosofía política

De manera general podemos caracterizar a la filosofía política como la rama de la filosofía que se ocupa de investigar la naturaleza, causas y efectos del *buen* y el *mal* gobierno [19]. Por supuesto, esta definición, más bien vaga, implica que hay por lo menos algo que nos permite distinguir los buenos gobiernos de los malos. Y esto nos garantiza que hay alguna suerte de conocimiento. En este contexto el concepto de *gobierno* es, ciertamente, mucho más incluyente que el positivo, pues admite una noción más amplia de institución política: gobierno, ley, etcétera. Así pues, el objetivo principal de la filosofía política consiste en generar conocimiento sobre instituciones políticas.

En la búsqueda de su objetivo la filosofía política ha desarrollado dos grandes clases de problemas que comprenden, de manera global, su esencia y acción: problemas *de facto* y de origen. Así, por ejemplo, el primer conjunto incluye problemas como el de la autoridad: ¿Qué derecho tiene alguien para obligar a otro a hacer algo en contra de su voluntad? ¿Por qué deberíamos seguir una ley que no se ajusta a nosotros? [19] El segundo grupo tiene problemas como: ¿Qué es el estado de naturaleza y cuál es el objetivo del contrato social? ¿Acaso

estos conceptos están bien justificados? ¿Cuál es el origen y el futuro de nuestro estado? En este trabajo –así como en la serie de estudios que proponemos– tendemos a interesarnos más por la segunda categoría de problemas y, de manera muy especial, por las nociones de estado natural y contrato social.

2.2. Algunas obras fundamentales

Varios trabajos clásicos y autores famosos ya han discutido estos temas. De hecho podemos trazar la tradición contractualista en la Grecia clásica con los sofistas, en la Europa Medieval es posible rastrear una especie de contrato en las formas del feudalismo, al tiempo que es fácil denotar a Grocio y Pufendorf como los teóricos más relevantes del contrato en el Renacimiento [8]. Sin embargo, son pocas las obras que tienen la maestría y la influencia de la *República* de Platón, del *Leviatán* [15] de Hobbes, del *Contrato Social* [22] de Rousseau, del *Segundo Tratado* [17] de John Locke, del *Manifiesto* [18] de Marx y Engels o de la magna *Teoría de la Justicia* [21] de Rawls. En este trabajo, no obstante, por cuestiones de espacio y por mor de los estudios, nos enfocamos únicamente en tres de ellos: el *Leviatán*, el *Contrato Social* y *Teoría de la Justicia*.

Ahora bien, esta elección no es arbitraria. Centramos nuestra atención en estas obras maestras no sólo porque son famosas, sino porque en ellas encontramos una muy buena (e.d., susceptible de ser abstraída) descripción del estado de naturaleza y las posibles explicaciones del origen del estado en términos formales y experimentales; y porque estas tres obras proveen argumentos clásicos para justificar –y en algunos casos explicar– qué forma de gobierno es la más óptima.

El estado de naturaleza es un estado hipotético que describe la vida humana antes de cualquier forma de gobierno factual. Sin embargo, a pesar de que estos estados están bien descritos, no hay ningún método formal o experimental que confirme o desapruebe tales ideas. Por otro lado, las posibles explicaciones del origen del estado, como sabemos, descansan sobre las premisas de un contrato social que, en este contexto, es una forma de auto-organización entre agentes, una convención entre individuos que renuncian a sus derechos naturales para convertirse en sujetos de derechos civiles. Esta transición, usualmente denominada *contrato*, de un estado de naturaleza a un estado civil es otra hipótesis que necesita alguna forma de confirmación además de su respectiva argumentación. Los resultados de este primer estudio, como veremos, nos dan pautas buscar posibles soluciones a este tipo de problemas.

Entonces comencemos por describir, rápidamente, las líneas generales de la descripción de Hobbes. En el *Leviatán* [15] Hobbes argumenta que en la ausencia de instituciones políticas que provean autoridad el estado de naturaleza es un estado de guerra y competencia constante porque, dado que los humanos son todos libres e iguales, se sigue que todos desean lo mismo y, por tanto, luchan por ello. Pero ya que no hay autoridad que provea una base para la confianza la conducta social característica está regulada, por decirlo así, por una libertad irrestricta que conlleva al egoísmo. Y por tanto, ya que no hay restricciones legales, el estado de naturaleza se define como un estado de *bellum omnium contra omnes*. Es sumamente importante notar que el estado de guerra constante parece ser una

consecuencia de las condiciones de igualdad y libertad, y no propiamente una consecuencia de una naturaleza que es de suyo malvada, como usualmente se suele explicar; pero aún así, la moralidad y la responsabilidad son prácticamente nulas. Este es el sentido preciso detrás del famoso *dictum* de Hobbes: *homo homini lupus est*. Sin embargo, de acuerdo al análisis antropológico hobbesiano, los humanos también poseen facultades cognitivas y, en su ejercicio, los humanos prefieren garantizar su seguridad a cambio de perder su libertad natural por medio de un contrato que otorga el poder a un gobernador.

Por el contrario, la visión de Rousseau es más bien una defensa del estado de naturaleza. En el *Contrato Social* [22] Rousseau propone que el estado de naturaleza, dado que no tiene influencia de ninguna institución política, es un estado de libertad natural y felicidad que se manifiesta en la igualdad de los individuos bajo la premisa de que todos somos naturalmente libres. Ya que ese es el estado natural del ser humano y, observa Rousseau, vivimos actualmente *–mutatis mutandis–* entre *cadena sociales*, hay una suerte de contradicción: si somos naturalmente libres, cómo es posible que vivamos entre tantas restricciones legales y morales. Esta contradicción sólo puede ser explicada por la existencia del gobierno actual que es la fuente de los problemas sociales, la inequidad y la pérdida de libertad. A pesar de cierta obscuridad, la meta de Rousseau es clara: no es la inocente reinvención del estado de naturaleza y el *buen salvaje* –en contraste con el *mal salvaje* de Hobbes, por decirlo así–, sino más bien la afirmación de que el gobierno debe considerar al estado de naturaleza como futuro y no sólo como pasado, como su destino y no sólo como su origen.

Por otro lado, en *Teoría de la Justicia* Rawls también solicita el recurso del hipotético estado de naturaleza, ya no como una hipótesis histórica (como en el caso de Hobbes y Rousseau), sino como una condición especial para un experimento mental. En una posición original en la que los individuos dispuestos a tomar partido en un contrato están en un velo de ignorancia –una condición que no permite a los contratantes saber con certeza el resultado de su elección–, se sigue que los agentes eligen, por *decisión racional*, dos principios: el principio de igualdad y el principio de diferencia. El experimento *gedanke* de Rawls va, más o menos, así: asumamos que tenemos dos individuos racionales en un momento previo a su llegada al mundo. Ambos están en un velo de ignorancia y se les presentan las siguientes alternativas: igualdad o desigualdad de libertades y oportunidades. Si ambos eligen la igualdad, seguramente al llegar al mundo tendrían justificación en exigir que las desigualdades fueran reparadas y las libertades fueran distribuidas convenientemente; por el contrario, si eligieran la desigualdad, no tendrían derecho para demandar reparación alguna en caso de quedar del lado de los desprotegidos (e.d., de los que no tienen el mismo número de libertades y oportunidades). Dado que ambos están en velo de ignorancia y son racionales, la estrategia más óptima consiste en elegir la igualdad.

Independientemente de si estamos de acuerdo con estas propuestas de filosofía política, lo que creemos, después de este breve resumen de argumentos, es que estos *modelos* de estado natural pueden ser, propiamente hablando, *modelados* computacionalmente usando simulación social basada en agentes, pues estas

propuestas ofrecen una caracterización de los individuos, del ambiente y de sus mutuas relaciones.

2.3. Simulación social basada en agentes

La simulación computacional consiste en el uso de modelos computacionales para ganar comprensión acerca de la conducta de sistemas complejos mediante modos de visualización y evaluación de diseños. En las simulaciones de esta clase el uso de ambientes artificiales es frecuentemente necesario porque, seguramente, el sistema a ser simulado no puede ser accesado directamente por razones observacionales (porque la simulación tiene que ser *ad hoc*), éticas (porque la seguridad de los individuos se vería comprometida) o prácticas (porque los costos de experimentación y adquisición de datos son muy altos) [2]. Los elementos principales de este tipo de simulación requieren:

- Un conjunto de *agentes* con sus diversas especificaciones.
- Un *ambiente* que proporcione a los agentes un *locus* para la percepción y la acción.
- Un mecanismo de *interacción* que permita la comunicación con el ambiente y entre agentes.

Esta clase de simulación puede ser utilizada como una herramienta experimental para el estudio de problemas de corte social [12,13] y, como se puede adivinar, consideramos que contribuye al tratamiento de las discusiones propias de la filosofía política y social. La idea general es que la simulación social basada en agentes puede mostrar una conducta análoga a la de las organizaciones humanas. Además se ha sugerido que tiene las siguientes ventajas [13]:

- Permite la observación de propiedades del modelo que en un principio pueden ser analíticamente derivables pero que no han sido establecidas aún.
- Provee condiciones para encontrar posibles alternativas a un fenómeno natural.
- Da oportunidades para estudiar propiedades que son difíciles de observar en la naturaleza.
- Facilita el estudio de fenómenos en aislamiento que pueden ser grabados y reproducidos si es necesario.
- Modela la sociabilidad explícitamente mediante las representaciones que tienen ciertos agentes de otros agentes.

Por tanto, no resulta complicado notar que podemos realizar estos estudios con una aproximación basada en agentes: las ideas de Hobbes, Rousseau y Rawls constituyen modelos clásicos de filosofía política que pueden ser modelados con tecnologías multi-agente para confirmar o falsear hipótesis, proveer respuestas a problemas aún abiertos y contribuir a la discusión filosófica.

3. El modelo

Ya que hemos asentado el problema y un posible curso de solución exponemos la especificación abstracta de los agentes y el ambiente. Para hacer esto usaremos un enfoque BDI (por *Beliefs*, *Desires* e *Intentions*). Aunque ciertamente existen otros modelos de agencia basamos nuestra especificación en este esquema no sólo por sus ventajas [11], sino también porque estas propuestas de filosofía política recurren a explicaciones en términos BDI. Pero también hacemos esta elección porque éste tiene una tradición filosófica patente en Searle [23], Dennett [9] y, principalmente, Bratman [5]; una sólida fundamentación formal en el ámbito de las ciencias computacionales [20] y hasta en el reino de los lenguajes de programación [4].

Pues bien, para este estudio hemos propuesto una tipificación de tres modelos clásicos de filosofía política. Y para lograr esto comenzamos en un primer momento siguiendo el patrón propuesto por la clasificación de Shook [24], que ha sugerido un interesante proyecto de filosofía política comparada –que no de política comparada. Siguiendo su taxonomía las teorías de filosofía política que hemos bosquejado pueden exponerse sintéticamente como en el Cuadro 1.

Tabla 1. Taxonomía general. Adaptada de [24]

Agente			
Categoría	Tipo Hobbes	Tipo Rousseau	Tipo Rawls
Humanidad	Obstinada	Simple	Neutral
Moralidad	Por principios	Intuitiva	Personal
Socialidad	Plural	Sectaria	Plural
Responsabilidad	Obediencia	Libre	Autónoma

Por supuesto, las palabras usadas por este modelo son reservadas y tienen un significado más o menos preciso. Así, bajo éste debemos entender por *Humanidad* al conjunto de características antropológicas que describen el estado de los agentes individuales. *Moralidad* indica cuál es el tipo de reglas que rigen el comportamiento de tales agentes individuales. Por *Socialidad* se entiende cuál es la manera en la que se organizan los agentes y, por último, *Responsabilidad* ha de pensarse como el modo en que se responde a la *Moralidad*. Además hay una quinta categoría –*Forma de gobierno*– que por el momento no consideraremos, aunque la retomaremos más adelante.

Una humanidad de naturaleza *Obstinada* queda definida, en términos coloquiales, como mala, aunque sería más preciso definirla como una humanidad egoísta. Por una *Simple* se entiende una naturaleza buena o altruista. Y una humanidad *Neutral* es aquella intermedia que no es obstinada ni simple. Por supuesto, estas rápidas narraciones no clarifican mucho las definiciones de estas variables, pero por el momento las dejaremos flotando en una idea general, si bien las precisaremos renglones abajo.

Por una moralidad *Por principios* se entiende a una moralidad que tiene que fundamentarse en reglas o normas de carácter heterónomo, por lo que la responsabilidad correspondiente consiste en la *Obediencia*. Una moralidad *Intuitiva* es aquella que no necesita justificarse con reglas o normas porque es, díganoslo así, innata y, por ello, su responsabilidad es enteramente *Libre*. Una moralidad *Personal* es la que se fundamenta en reglas o normas que se eligen autónomamente, y así su responsabilidad es *Autónoma*. Las variables de socialidad se explican como usualmente se entienden el pluralismo y el sectarismo.

Ahora, si bien este modelo nos permite pensar en términos de una línea de investigación de filosofía política comparada, hay algunos aspectos del mismo que no son tan precisos como quisiéramos y no sirven para solventar nuestro problema. Entonces, para preservar las ventajas de éste y soslayar su falta de precisión, adaptamos el de Bunge [7] donde estos tres tipos de agentes se pueden tipificar finamente como veremos a continuación. De acuerdo a este otro modelo hay tres tipos de comportamiento socio-político (altruista, egoísta y reciprocador) que pueden ser capturados mediante la definición de un compromiso social C :

$$C = r + e * s + m$$

donde $e, m \in [0, 1]$, $r, s \in \mathbb{N}$ y decimos que r denota el retorno esperado de una acción, e es un factor de equidad, m es un grado de moralidad y s es un valor de socialidad. Entonces en los casos arquetípicos tenemos que:

- Si $m = e = 0$, $C = r$; e.d., que el egoísmo ocurre cuando el compromiso social es igual al retorno esperado.
- Si $m = e = 1$, $C \geq r$; e.d., que el altruísmo ocurre cuando el compromiso social es mayor o igual que el retorno esperado.
- Si $e \geq 0$ y $m > 0$, $C > r$; e.d., que la reciprocación ocurre cuando el compromiso social es mayor que el retorno esperado.

Con esta información no es tan difícil observar cómo se pueden aparear estos dos modelos: el primero, para proveer la clasificación y la interpretación de las propuestas de filosofía política; el segundo, para dotar de precisión y formalidad al primero. De esta manera lo que sugerimos es que la variable m puede interpretarse en términos de las variables de *Humanidad* y *Moralidad*, la equidad e bajo la variable de *Responsabilidad* y la socialidad s como el número de interacciones entre agentes.

Si seguimos las argumentaciones clásicas, como la moralidad y la responsabilidad en Hobbes tienen que estar, respectivamente, dadas por principios y por obediencia, en un estado de naturaleza tienen que valer originalmente 0. En el caso de Rousseau, como éstos son simples e intuitivos, en un estado de naturaleza tienen que valer 1; por el contrario, en Rawls, como hablamos de un carácter neutral y personal, asumimos que la moralidad es diferente de 0 y la equidad es mayor o igual a 0, en particular asumiremos el promedio de los otros dos (Cuadro 2). Y, por mor del experimento, r quedará definido mediante un pago que se especificará más abajo.

Tabla 2. Instanciación del modelo

Agente			
Variable	Tipo Hobbes	Tipo Rousseau	Tipo Rawls
r	Pago	Pago	Pago
m	0	1	> 0
e	0	1	≥ 0

Por tanto, ya con la adaptación de ambos modelos, lo que tenemos es lo siguiente:

- Cuando $m = e = 0$, $C = r$; e.d., los agentes tipo Hobbes se comportan como egoístas.
- Cuando $m = e = 1$, $C \geq r$; e.d., los agentes tipo Rousseau se comportan como altruistas.
- Cuando $e \geq 0$ y $m > 0$, $C > r$; e.d., los agentes tipo Rawls se comportan como reciprocadores.

Además sugerimos el siguiente ambiente definido mediante una matriz de pagos basada en el Dilema del Prisionero tradicional (Cuadro 3). La idea de esta matriz es la siguiente: dados dos agentes, si ambos cooperan, ambos ganan 3 puntos, esto es, $r = 3$ para cada uno de ellos. Si ambos no cooperan, $r = 2$ para ambos. Si uno coopera y el otro no lo hace, el que no coopera gana 5 puntos (fenómeno denominado como *temptation to deffect*) y el que coopera obtiene 0 puntos (*sucker's payoff*).

Tabla 3. Matriz de pagos basada en el Dilema del Prisionero

Ambiente		
Pago	Cooperación	No-Cooperación
Cooperación	3,3	5,0
No-Cooperación	0,5	2,2

Con estos componentes tenemos una caracterización general de los agentes y el ambiente en el que se desarrollarán para tratar un problema de filosofía política. Ahora exponemos los experimentos y los resultados.

4. Experimentos y resultados

Hemos diseñado 4 experimentos. Cada uno cuenta con un conjunto de agentes y un ambiente implementado en Jason [4]. Brevemente, los experimentos son como siguen:

- Experimento *I*: 5 tipo Hobbes vs 5 tipo Rousseau.
- Experimento *II*: 5 tipo Rousseau vs 5 tipo Rawls.

- Experimento *III*: 5 tipo Hobbes vs 5 tipo Rawls.
- Experimento *IV*: 5 tipo Hobbes vs 5 tipo Rawls vs 5 tipo Rousseau.

Si todo sale en orden, lo que esperaríamos ver debería estar descrito por las siguientes afirmaciones términos de ganancias (e.d., en términos de pagos o de la variable r):

- *I*: Los agentes Hobbes ganan más que los Rousseau.
- *II*: Los agentes Rousseau ganan más o menos lo mismo que los Rawls.
- *III*: Los agentes Hobbes ganan más o menos lo mismo que los Rawls.
- *IV*: En un escenario de todos contra todos, el grado de ganancias va creciendo en el siguiente orden: Rousseau, Rawls, Hobbes.

Pero deberíamos observar lo siguiente en términos de compromiso social (C):

- *I*: Los agentes Hobbes tienen menor compromiso que los Rousseau.
- *II*: Los agentes Rousseau tienen más o menos el mismo compromiso que los Rawls.
- *III*: Los agentes Hobbes tienen más o menos el mismo compromiso que los Rawls.
- *IV*: En un escenario de todos contra todos, el grado de compromiso social va decreciendo en el siguiente orden: Rousseau, Rawls, Hobbes.

De un promedio de 5 corridas del experimento *IV*, de 2000 iteraciones cada una, se han obtenido los promedios. Entonces lo que observamos es que el compromiso social se comporta como si fuera el simétrico del pago, en efecto, lo que podemos ver es lo siguiente: a mayor ganancia, menor compromiso; e inversamente, como se muestra en la Figura 1.

4.1. Discusión

Claro, en términos de ganancias ya hay varias investigaciones que reportan que las estrategias ALLD (egoístas o *all defect*) y ALLC (altruistas o *all cooperate*) y Tit-4-Tat (reciprocadoras o *quid pro quo*) se comportan como se supone que lo hagan [1,3]; pero no explican las causas o efectos de la contraparte de las ganancias: el compromiso social entre agentes.

Ya que podemos notar que:

- En agentes tipo Rousseau: $C = \text{pago} + 1 * \text{iteraciones} + 1$
- En agentes tipo Rawls: $C = \text{pago} + 0,5 * \text{iteraciones} + 1$
- En agentes tipo Hobbes: $C = \text{pago}$

Argumentamos que el elemento que pesa no es tanto el de las ganancias, sino el valor del producto de las iteraciones por la equidad, esto es, en si las relaciones sociales son totales, nulas o diferentes de cero. Por tanto, sugerimos que el factor que explica este balance es la socialidad. Y resulta interesante observar que los agentes tipo Rawls (reciprocadores) muestran una buena relación

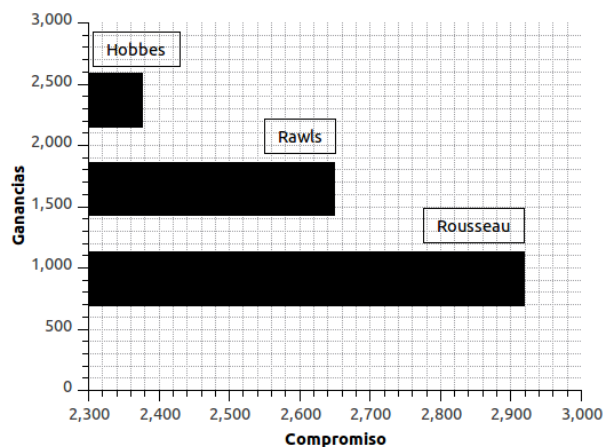


Figura 1. Promedios del experimento IV

entre compromiso social y ganancia; mientras los agentes Rousseau (altruistas) muestran una tendencia alta al compromiso social, pero con poca ganancia; y los agentes Hobbes muestran una tendencia alta a la ganancia, pero poco compromiso. Estas observaciones resultan relevantes si tomamos en cuenta que la noción de justicia como equidad (*fairness*) es parte de la propuesta de Rawls. Pero más que relevante, es importante notar que esta propuesta provee razones para justificar una particular forma de gobierno que incluye las nociones de un estado de bienestar (*Welfare state*), mientras la propuesta de Rousseau justifica un estado de corte comunal y la de Hobbes la necesidad de un estado de carácter monárquico.

Ahora, en lo que concierne a los objetivos más generales de esta serie de estudios: qué podemos decir sobre el problema de qué teoría clásica es más plausible para explicar el origen del estado. Pues bien, sin duda parece que la propuesta de Rousseau, por ganar en compromiso social, parecería la más plausible para explicar el origen del estado. Sin embargo, no hay estado sustentable sin competencia. En cuestión de competencia, la propuesta de Hobbes es sin lugar a dudas ganadora, pero tanta competencia está condenada al fracaso, pues no hay estado sustentable sin integración. Si buscáramos un balance entre ganancias y compromiso social, lo que en la literatura inglesa se conoce como *trade-off*, la propuesta de Rawls resulta más plausible, lo cual es algo que se espera, pero las causas de este balance no han de buscarse en las ganancias o en la moralidad de los individuos, sino en la naturaleza y el tipo de socialidad.

5. Conclusion

En este punto nos gustaría mencionar, brevemente, que la importancia de las respuestas a estos problemas proviene no sólo de un interés en el pasado sino, de

manera más importante, de un interés en el futuro de la conducta socio-política, pues consideramos que, por un lado, su carácter experimental enriquece el aspecto filosófico de la discusión –al proveer evidencia cuantitativa sobre cuestiones sobre las que usualmente se recurre a la opinión–, pero su lado filosófico proporciona alcances que van más allá de lo meramente computacional o experimental–al sugerir argumentaciones e interpretaciones para explicar hechos que observamos ordinariamente.

Finalmente, como parte de estudios futuros podemos comentar que estamos tratando con dos en particular: el modelado de otras filosofías políticas para continuar con las investigaciones de filosofía política comparada (por ejemplo, ¿Qué resultados pueden arrojar los modelos de Platón (sociedad de castas), Tomás de Aquino (sociedad jerarquizada y cristiana), Locke y Mill (sociedades democráticas y liberales)?) y la implementación de una conducta más compleja en ambientes más complejos en términos de autómatas celulares sin aprendizaje o con aprendizaje para tomar en cuenta los problemas tradicionales de adaptación [16].

Agradecimientos. El autor agradece a los revisores anónimos por sus comentarios y precisas correcciones; y a todos los apoyos por parte del Departamento de Artes y Humanidades y de la Coordinación de la Escuela de Filosofía de la Universidad Popular Autónoma del Estado de Puebla.

Referencias

1. Axelrod, R.: *The Evolution of Cooperation*. New York: Basic Books (1984)
2. Bandini, S., Manzoni, S., Vizzari, G.: Agent Based Modeling and Simulation: An Informatics Perspective. *Journal of Artificial Societies and Social Simulation* 12 (4) 4 (2009) <http://jasss.soc.surrey.ac.uk/12/4/4.html>
3. Bazzan, A., Bordini, R. H., Campbell, J. A.: Agents with Moral Sentiments in an Iterated Prisoner’s Dilemma Exercise. *Proceedings of the AAAI Fall Symposium on Socially Intelligent Agents*, 8–10 November (1997)
4. Bordini, R.H., Hübner, J.F., Wooldridge, M.: *Programming Multi-Agent Systems in AgentSpeak using Jason*. Wiley, England (2007)
5. Bratman, M.: *Intention, Plans, and Practical Reason*. Harvard University Press, Cambridge (1987)
6. Bunge, M.: *Diccionario de filosofía*. México, Siglo Veintiuno Editores (2002)
7. Bunge, M.: *Political philosophy*. Transaction Publisher (2009)
8. Copleston, F.: *A History of Philosophy* (9 volumes reissued). New York: Image Books, (1993–1994)
9. Dennett, D.: *The Intentional Stance*. MIT Press, Cambridge, MA., (1987)
10. Genesereth, M.R., Nilsson, N.J.: *Logical Foundations for Artificial Intelligence*. Morgan Kauffman Publishers, Inc., Palo Alto, CA., USA, (1987).
11. Georgeff, M., Pell, B., Pollack, M., Tambe, M., and Wooldridge, M.: The Belief-Desire-Intention Model of Agency. In: J. P. Muller, M. Singh, and A. Rao, editors *Intelligent Agents V Springer-Verlag Lecture Notes in AI Volume 1365*, March (1999)

12. Gilbert, N., Doran J (eds): *Simulating Societies: the Computer Simulation of Social Phenomena*. UCL Press: London (1994)
13. Gilbert, N. Conte, R. (eds): *Artificial Societies: the Computer Simulation of Social Life*. UCL Press: London (1995)
14. Goodin, R.E. Pettit, P. (eds): *Contemporary political philosophy An anthology*. Blackwell Publishers Ltd. (1997)
15. Hobbes, Th.: *Leviathan*. ed Edwin Curley, Hackett, Indianapolis, (1994 [1651/1668]).
16. Kinny, D., Georgeff, M.: Commitment and effectiveness of situated agents. In: *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91)*, 82–88, Sydney, Australia (1991).
17. Locke, J.: *The Two Treatises of Civil Government*. Hollis ed. (1689)
18. Marx, K., Engels, F.: *Selected Works, Vol. One*. Progress Publishers, Moscow (1969)
19. Millner, D.: *Political philosophy. A Very Short Introduction*. Oxford University Press (2003).
20. Rao, A.S., Georgeff, M.P.: Modelling Rational Agents within a BDI-Architecture. In: Huhns, M.N., Singh, M.P., (eds.) *Readings in Agents*, pp. 317–328. Morgan Kaufmann (1998)
21. Rawls, J.: *A Theory of Justice*. Harvard University Press, Cambridge MA (1971/1999)
22. Rousseau, J.J.: *The Basic Political Writings*. Trans. Donald A. Cress. Hackett Publishing Company (1987)
23. Searle, J.R.: *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, England (1969)
24. Shook, J.R.: Comparative political philosophy categorizing political philosophies using twelve archetypes. *Metaphilosophy* Vol. 40, No. 5 0026-1068 (2009)

Manejo de la cadena de suministro mediante un sistema con múltiples agentes colaborativos

Carlos R. Jaimez-González

Departamento de Tecnologías de la Información, Universidad Autónoma Metropolitana - Cuajimalpa, Av. Constituyentes No. 1054, Col. Lomas Altas, C.P. 11950, México D.F.
cjaimez@correo.cua.uam.mx

Resumen La forma de hacer negocios ha cambiado significativamente en las últimas décadas. La gran cantidad de información y las nuevas tecnologías incrementan las expectativas de los clientes con respecto a los productos y servicios que las compañías ofrecen. El Internet ha propiciado el cambio de los procesos de negocio individuales, hacia un modelo de negocios más distribuido, colaborativo, en un ambiente de comercio electrónico, y que se adapta a las condiciones de mercado competitivas y cambiantes. Este artículo presenta un sistema de toma de decisiones, el cual está basado en múltiples agentes colaborativos, explora diferentes estrategias del proceso de producción en su conjunto, y ofrece soluciones para manejar cadenas de suministro en ambientes de comercio electrónico distribuidos. El sistema está diseñado para soportar diferentes tipos de interfaces, que permitan interactuar con los modelos de negocio ya existentes, a través del manejo e integración de la cadena de suministro cooperativa.

Palabras clave: Sistemas multi-agente, comercio electrónico, manejo de la cadena de suministro, tecnología de agentes, agentes colaborativos.

1. Introducción

La forma de hacer negocios ha cambiado significativamente en las últimas décadas. La gran cantidad de información y las nuevas tecnologías incrementan las expectativas de los clientes con respecto a los costos y servicios, así como también la competencia global hace que los líderes en las empresas busquen nuevas formas de abordar los negocios. Ya no están en posición de confiar en estrategias de negocio estáticas, sino que deben ser capaces de enfrentarse a ambientes que cambian rápidamente y que en algunas ocasiones son inciertos, tales como los tipos de cambio bancarios, los cambios en las situaciones políticas, el retraso en la entrega de materiales para producción, las rupturas con proveedores, la falla en instalaciones de producción, la ausencia de trabajadores, la cancelación o cambio de órdenes de clientes, etc.

Las compañías deben llevar a cabo una serie de actividades, tales como el abastecimiento de materiales, la fabricación de productos, el almacenamiento de productos, las ventas y entregas de dichos productos, los servicios al cliente, entre otras. Todas estas actividades deben llevarse a cabo como si fueran un proceso dinámico, de tal forma que se mantenga un balance entre ellas. En una empresa, ésta es precisamente

la tarea principal del “*Supply Chain Management (SCM)*” o “*Manejo de la Cadena de Suministro*”; el cual se encarga de negociar con proveedores para obtener los materiales necesarios para producción, manejar órdenes de los clientes, controlar el inventario, establecer tiempos para la fabricación y entrega de productos, etc.

Tomando en consideración la globalización del mercado, es común que las empresas tengan negocios distribuidos, donde los proveedores y los clientes se encuentran dispersos alrededor del mundo. El desarrollo de las tecnologías de la información ha propiciado que las organizaciones usen el Internet para participar en el comercio electrónico; de esta forma reducen sus costos administrativos y transaccionales, incrementan sus utilidades, e interactúan con un mayor número de socios comerciales en diferentes ubicaciones geográficas. El Internet ha propiciado el cambio de los procesos de negocio individuales, hacia un modelo de negocios más distribuido y colaborativo. Para ser capaces de manejar este modelo, las empresas necesitan una solución, que les permita participar en ambientes de comercio electrónico. Esta solución debe incluir un tipo de sistema que ayude en la toma de decisiones y al mismo tiempo se adapte a los cambios; el cual pueda recabar y procesar información de un gran número de fuentes heterogéneas, así como ayudar a tomar decisiones precisas en condiciones de mercado competitivas y cambiantes. Debido a la necesidad de diseñar estrategias para coordinar e integrar entidades de negocios dentro de estos ambientes, uno de los objetivos de este trabajo es el desarrollo de estrategias con tecnología de agentes para comercio electrónico. Para su implementación y pruebas, es necesario el desarrollo de un sistema para manejar cadenas de suministro en los ambientes mencionados.

Este artículo presenta un sistema de toma de decisiones, el cual está basado en múltiples agentes colaborativos, explora diferentes estrategias del proceso de producción en su conjunto, y ofrece soluciones para manejar cadenas de suministro en ambientes de comercio electrónico distribuidos. El sistema está diseñado para soportar diferentes tipos de interfaces, que permiten interactuar con los modelos de negocio ya existentes de otros participantes, a través del manejo e integración de la cadena de suministro cooperativa. Debido a que el mercado electrónico es un modelo de negocios recientemente establecido y es concebido como una actividad de resolución de problemas distribuido cooperativo, el diseñar sistemas para el manejo de la cadena de suministro se ha vuelto más importante que nunca.

El resto del artículo está organizado de la siguiente manera. En la siguiente sección se proporcionan los antecedentes y estado del arte del SCM, su evolución, el enfoque multi-agente, y una plataforma de experimentación. La sección 3 presenta la arquitectura del sistema, y sus seis agentes colaborativos. Algunas estrategias de ventas y producción para agentes de comercio electrónico, son presentadas en la sección 4. Finalmente, se proporcionan conclusiones y trabajo futuro en la sección 5.

2. Antecedentes y estado del arte

Esta sección proporciona algunos antecedentes, y resume el estado del arte en el campo del manejo de la cadena de suministro (SCM) y los sistemas de toma de decisiones para SCM. En particular, la tendencia en estas áreas es moverse de los proce-

tos de negocio estáticos, hacia modelos distribuidos y dinámicos. Los sistemas de toma de decisiones para SCM son actualmente diseñados como sistemas multi-agentes, para soportar dichos modelos. Muchos grupos de investigación alrededor del mundo dedican su trabajo a explorar varios problemas dentro del dominio de SCM, y han llevado a cabo estudios sobre la plataforma de simulación de TAC SCM, la cual es utilizada ampliamente en investigación, y es descrita al final de esta sección.

2.1. Evolución de SCM

SCM es un proceso complejo, el cual incluye una variedad de actividades interrelacionadas, tales como negociación con proveedores para obtener materiales, competencia por ganar órdenes de clientes, manejo de inventario, programación de producción, entrega de productos a clientes. Los conceptos de SCM han sido utilizados por compañías desde principios del siglo XX, inclusive en la literatura se encuentran discusiones desde los años 50 en [1]. En los 80's, la idea de automatizar procesos de negocio a través de SCM se hizo muy popular, sin embargo los expertos trataron cada una de las entidades de la cadena de suministro como un proceso estático aislado del resto. Muchos trabajos de finales de los 90's aún se concentraban en resolver solamente subáreas separadas de SCM [2], [3]. En [4], se empieza a ver el problema de SCM como un ambiente dinámico, y como un proceso integrado con restricciones [5].

Con el surgimiento del WWW, los sistemas de comercio electrónico se han hecho extremadamente populares, principalmente en la última década. Ya se tienen modelos de comercio electrónico integrados, los cuales incluyen a proveedores, clientes, socios comerciales, agentes de comercio electrónico, etc., dentro de un ambiente electrónico global [6]. Recientemente se han propuesto arquitecturas de sistemas para soportar la participación en el comercio electrónico [7]. Muchos investigadores coinciden en que la arquitectura de un sistema de toma de decisiones que soporte SCM debe ser ágil para combatir con la dinámica de los mercados electrónicos, así como también debe ser fácilmente reconfigurable, adaptarse a varios negocios, y soportar varios protocolos de diferentes ambientes comerciales. El enfoque multi-agente ha demostrado ser el más adecuado para cubrir estos requerimientos [8], [9], [10], [11].

2.2. SCM basado en el enfoque multi-agente

Es un hecho que la tecnología de agentes se ha convertido en la herramienta más popular para diseñar sistemas distribuidos para el manejo de la cadena de suministro, ya que proporciona una forma dinámica y adaptable para manejar por separado cada uno de los eslabones de la cadena. A diferencia de los sistemas centralizados, los sistemas de manejo de la cadena de suministro basados en agentes pueden responder rápidamente a los cambios internos o externos a través de la toma de decisiones. En [12] se proporciona una revisión detallada de algunos sistemas basados en agentes, diseñados e implementados con fines industriales.

Otra ventaja de diseñar la solución SCM como un sistema multi-agente es que nos permite separar las diferentes tareas del SCM, explorarlas de manera independiente, y analizarlas en su conjunto. Esta característica es particularmente importante, ya que

nos permite enfocarnos de manera separada en varios aspectos de la cadena de suministro: la parte de la demanda, la cual se refiere a la venta de productos a los clientes; y la parte de la producción, la cual involucra la obtención de los materiales y la fabricación de los productos. En el caso de los vendedores, el principal problema al que se enfrentan cuando manejan sus cadenas de suministro es decidir qué ofertas harán a sus clientes, qué precios les darán, cuándo y a quién vendérselos, etc.; todo esto es con el objetivo de incrementar su utilidad. La tarea no es fácil de resolver en el contexto del comercio electrónico, donde los precios se establecen dinámicamente.

Es importante señalar que se han propuesto varios enfoques para diseñar sistemas de toma de decisiones para SCM basados en agentes [10], y se han desarrollado arquitecturas para comercio electrónico orientadas a agentes [6]. Una descripción de proyectos recientes que utilizan agentes en el contexto de SCM es proporcionada en [13].

2.3. *Trading agent competition SCM*

Dado que es necesario llevar a cabo una serie de experimentos para probar las estrategias de producción y ventas en la cadena de suministro, las empresas reales no ponen en riesgo sus negocios para probar las soluciones propuestas. Para satisfacer esta demanda, se han llevado a cabo varios intentos de crear herramientas de simulación para SCM, que permitan el estudio y experimentación de diferentes algoritmos. Entre estos intentos, destaca el juego denominado TAC SCM [14], el cual es probablemente la mejor plataforma de experimentación para realizar pruebas de sistemas SCM, ya que encapsula muchos de los problemas que pueden ser encontrados en ambientes SCM reales: restricciones de tiempo, oponentes impredecibles, capacidad de producción limitada, etc. Para la experimentación del sistema propuesto, se utilizará esta plataforma, tanto para pruebas de desempeño de la solución del sistema SCM completo, como de las estrategias de producción y ventas implementadas.

Este juego fue diseñado por la Carnegie Mellon University y el Swedish Institute of Computer Science (SICS) en el 2003 como parte de la competencia internacional de agentes de comercio (International Trading Agent Competition) [15]. Este juego permite la competencia de agentes de software en el contexto de SCM, para evaluar sus diferentes propuestas y algoritmos, los cuales buscan diseñar sistemas de toma de decisiones que sean exitosos. Los agentes que participan en esta competencia son desarrollados por diferentes grupos de investigación alrededor del mundo.

En TAC SCM seis agentes compiten entre ellos en una simulación de 220 días, la cual tiene una duración real de 55 minutos. Cada agente es un fabricante que ensambla computadoras, a partir de sus componentes básicos: CPUs, tarjetas madre, memorias RAM, y discos duros. Los CPUs y las tarjetas madre están disponibles en dos familias de productos diferentes: IMD y Pintel. Un CPU Pintel solamente funciona con una tarjeta madre Pintel, mientras que un CPU IMD puede ser incorporado solamente a una tarjeta madre IMD. Los CPUs están disponibles en dos velocidades, 2.0 y 5.0 GHz; las memorias se fabrican en dos tamaños, 1 y 2GB; y los discos duros en dos tamaños, 300 y 500GB. Existen en total diez componentes diferentes, los cuales pueden ser combinados, para tener un total de 16 modelos de computadoras diferentes. Los agentes necesitan obtener estos componentes de 8 proveedores diferentes. Por

otro lado, los agentes necesitan asegurar órdenes de clientes todos los días, para las cuales es necesario ensamblar el número y modelo de computadora requerido en su fábrica, y subsecuentemente entregar los productos terminados. Todos los agentes comienzan el juego sin dinero en sus cuentas bancarias, sin componentes o computadoras fabricadas en su inventario, y sin órdenes de clientes pendientes de entregar. Cada modelo de computadora requiere un número diferente de ciclos para que pueda ser fabricado, y un agente tiene una capacidad limitada de 2000 ciclos de producción al día. El servidor TAC simula a los proveedores, clientes y banco, y proporciona servicios de producción y almacenamiento a cada uno de los 6 agentes que participan. El agente que logre hacer la mayor utilidad al final del juego es el ganador. Las reglas de operación del juego, así como el software requerido para realizar experimentación sobre esta plataforma pueden ser descargados directamente de [15].

3. Arquitectura del sistema

En esta sección se describe la arquitectura del sistema y su comportamiento. El sistema será desarrollado utilizando los principios de los sistemas multi-agente. De esta forma, varios problemas que son encontrados en el dominio de SCM pueden ser sistemáticamente estudiados. Aunque está planeado que el sistema compita en el juego organizado por TAC SCM, la arquitectura del mismo será genérica y reconfigurable, de tal forma que pueda ser adaptada a ambientes similares, así como también pueda ser modificada para cumplir los requerimientos particulares de un negocio específico. Asegurándose que se tenga un sistema multi-agente para SCM, se podrán explorar diferentes problemas de la cadena de suministro por separado. Asimismo el sistema propone la utilización de una arquitectura basada en servicios Web, mediante la cual se permita la convivencia de otros sistemas (agentes) desarrollados con otras tecnologías u otros lenguajes de programación.

En un ambiente comercial, el sistema debe producir ciertas decisiones en la adquisición de materiales, ventas, producción, y entrega de los productos. Para tomar esas decisiones, se deben completar varias tareas, las cuales pueden depender del resultado de otras o ser independientes. Debido a que hay restricción de tiempo para la toma de decisiones, las tareas que son independientes deben de realizarse en paralelo para ahorrar tiempo. El abordar el problema desde una perspectiva basada en agentes es adecuada para cumplir estos requerimientos, y es por ello que sistemas de toma de decisiones para el manejo de la cadena de suministro, son diseñados como sistemas multi-agente. A través de la coordinación y colaboración, los agentes son capaces de manejar las actividades distribuidas a lo largo de la cadena de suministro. El enfoque multi-agente es *“una forma natural de modularizar sistemas complejos”* [13], tal como los sistemas para el manejo de la cadena de suministro.

El sistema consiste de seis agentes, cinco de ellos corresponden a cada una de las entidades en la cadena de suministro: Agente de Demanda, Agente de Suministro, Agente de Inventario, Agente de Producción, y Agente de Entrega; el sexto agente es el Agente Coordinador, el cual se encarga de coordinar el desempeño del sistema, así como también se comunica con el ambiente exterior. Teniendo agentes separados nos

permite la reutilización del sistema por partes, y experimentar con ellas en un amplio espectro de aplicaciones SCM reales, ya que cada agente puede ser fácilmente incluido o quitado del sistema sin arruinar el desempeño del sistema completo. Aunque cada agente tiene sus propias metas, los agentes trabajan en cooperación para alcanzar la meta global común: maximizar la utilidad total. Esta meta, a su vez puede ser dividida en submetas: maximizar la utilidad en ventas, minimizar los precios de compra de materiales, minimizar los costos de almacenamiento de materiales, y minimizar las penalizaciones por entregas tardías.

3.1. Agentes colaborativos

En esta subsección se proporciona una breve descripción de los seis agentes que son utilizados en el sistema.

Agente Coordinador. Es responsable de la comunicación con el ambiente externo (en este caso es el responsable de estar en contacto con el servidor TAC SCM), y de coordinar al resto de los agentes. En particular, este agente tiene las siguientes responsabilidades: actualizar el inventario; actualizar el status de la cuenta bancaria; recibir las ofertas de los proveedores; recibir las solicitudes y órdenes de los clientes; enviar ofertas a los clientes; enviar solicitudes y órdenes a los proveedores; compartir las agendas de producción y entrega de productos con otros agentes; recibir reportes del mercado y de precios; llevar un registro de solicitudes, ofertas, órdenes, agendas de producción y entrega de productos, reportes, y otra información compartida por los otros agentes; y coordinar el desempeño del sistema completo.

Agente de Demanda. Se encarga de las ventas de productos a los clientes. Diariamente recibirá las solicitudes y órdenes de los clientes del *Agente Coordinador*. Para cada una de las solicitudes de los clientes, el agente decide el precio que se le ofrecerá al cliente, mediante la predicción de precios. En la siguiente sección se consideran algunas estrategias que pueden ser implementadas para la predicción de precios de órdenes para los clientes. Este agente debe estar en comunicación con el *Agente de Producción*, para determinar si la fabricación de productos será suficiente para satisfacer las órdenes futuras de los clientes. La meta del *Agente de Demanda* es maximizar la utilidad a partir de las órdenes de los clientes.

Agente de Suministro. Este agente se encarga de generar solicitudes de materiales a los proveedores, considerando su demanda, el nivel actual del uso de los materiales, y el stock disponible. Este agente debe de utilizar estrategias para enviar solicitudes de materiales, y hacer predicciones para garantizar que exista suficiente stock, para que el *Agente de Producción* sea capaz de fabricar productos a partir de los materiales disponibles. Este agente es encargado de realizar un análisis de los precios de los materiales y determinar cuándo ordenarlos para minimizar el precio que se paga por ellos; este análisis puede estar basado en los precios pagados recientemente, los pre-

cios actuales, y los precios proporcionados en el reporte del mercado. Una vez que el *Agente de Suministro* recibe ofertas de parte de los proveedores, está en posición de generar una orden.

Agente de Inventario. Se encarga de la llegada de materiales de los proveedores; de la llegada de los productos terminados enviados por el *Agente de Producción*; del envío de materiales para fabricar productos; y del envío de productos terminados a los clientes. El *Agente de Inventario* lleva un registro de los materiales y productos solicitados por los *Agentes de Producción y de Entrega*, respectivamente, e intenta evitar que el inventario caiga por debajo de un límite establecido, para poder satisfacer las demandas. Para minimizar los costos de almacenamiento, el agente debe de ajustar el límite de materiales y productos de manera dinámica. Este agente se encuentra en comunicación con los *Agentes de Producción y de Suministro*, a través del *Agente Coordinador*, para mantener el stock disponible de productos y materiales.

Agente de Producción. Es responsable de programar la producción actual y proyectar la producción hacia el futuro. Dado que tiene información acerca de las solicitudes y órdenes de los clientes del *Agente de Demanda*, y acerca del stock de materiales del inventario del *Agente de Inventario*, este agente puede programar su producción para satisfacer a sus clientes. Para intentar maximizar la utilidad obtenida, debe considerarse que cada agente tiene una capacidad de producción limitada. Diariamente el agente programa las órdenes de producción dependiendo de sus fechas de entrega, utilidad y disponibilidad de materiales en stock, para posteriormente realizar más solicitudes de materiales si son necesarias (mediante el *Agente de Suministro*).

Agente de Entrega. La tarea de este agente es la entrega de los productos a los clientes, de acuerdo a las órdenes. Para evitar penalizaciones por entregas tardías, este agente programa la entrega de órdenes activas tan pronto como los productos estén listos por parte de la producción. Este agente se encarga de revisar las órdenes de entrega, las ordena por fecha, y realiza las entregas conforme los productos estén disponibles en inventario.

4. Estrategias

Esta sección describe las estrategias que se utilizarán en el sistema. La investigación de estrategias se enfoca en la parte de la demanda (estrategias de ventas) y en la parte de la fabricación de productos (estrategias de producción).

4.1. Estrategias de ventas

La generación dinámica de precios se ha convertido en un concepto importante que regula las relaciones entre vendedores y sus clientes, en los ambientes de comercio electrónico. Las estrategias de ventas con listas de precios fijos para todos los clientes

ya no funcionan, ya que los clientes tienen la posibilidad de comparar precios en minutos utilizando diferentes sitios Web. Los vendedores tienen que ser capaces de reaccionar inmediatamente a los cambios en la situación del mercado, incluyendo cambios en los volúmenes de demanda, las estrategias de sus competidores, y las preferencias de los clientes; también deben de tomar en cuenta sus volúmenes de stock, sus capacidades de fabricación de productos, así como las relaciones con sus proveedores. Estos factores, y algunos otros producen incertidumbre en el proceso comercial. Las subastas en línea han demostrado ser los mecanismos de generación dinámica de precios más eficientes, que permiten a vendedores y a compradores llegar a un acuerdo en los precios cuando participan en actividades de comercio electrónico.

Existen diferentes tipos de subastas en línea que definen diferentes protocolos de negociación entre vendedores y compradores; y uno de los problemas más discutidos acerca de ellas es la determinación del ganador de la subasta. Algunos ejemplos de soluciones que se han propuesto para resolver este problema pueden ser encontrados en [17] y [18]. En particular, se busca atacar el problema de predecir los precios que serán dados a las órdenes de los clientes, utilizando el tipo de subastas conocidas como "*first price sealed bid reverse auctions*", en las cuales la apuesta de cada apostador es enviada en sobre cerrado, y la apuesta con el menor precio gana la subasta. En el contexto de SCM, el siguiente escenario se lleva a cabo: un número determinado de fabricantes ofrecen al cliente sus precios del producto que solicitó; sin saber entre ellos los precios que cada uno ofrece al cliente. El cliente hace la orden al fabricante que le ofrece el menor precio. La capacidad de un fabricante para predecir el precio más bajo propuesto por sus oponentes es crucial, ya que así se puede tener una estrategia exitosa que les ayude a maximizar su utilidad.

Las estrategias que serán exploradas e implementadas son descritas brevemente a continuación. Una estrategia es predecir la probabilidad de que el precio ganador de la subasta esté en un intervalo determinado, y realizar una apuesta de acuerdo al precio más probable. Otra estrategia es predecir los precios basados en los detalles de las solicitudes de los clientes, la situación del mercado, y los resultados de subastas previas, y apostar de acuerdo al precio de la predicción. La tercera estrategia consiste en predecir los precios más alto y más bajo de las órdenes de los clientes, para cada producto, basados en una serie de tiempo para esos precios, y apostar entre los valores de la predicción. Otra estrategia es modelar el comportamiento del competidor y los precios que ofrece, y apostar justo por debajo de ellos. Para llevar a cabo lo anterior se propone utilizar técnicas estadísticas y de aprendizaje, tales como redes neuronales y programación genética, para intentar resolver el problema de predicción de precios. A diferencia de los métodos tradicionales de análisis técnico y estadístico, los métodos de aprendizaje pueden reaccionar a irregularidades del mercado más exitosamente, y así proporcionar resultados más adecuados, en condiciones de un SCM dinámico.

4.2. Estrategias de producción

Uno de los grandes problemas dentro de la cadena de suministro es determinar cuándo ordenar materiales, para fabricar los productos que han sido solicitados por los clientes. Por ejemplo, si un fabricante ordena materiales para que lleguen en un futuro

lejano, esto trae el beneficio de la reducción de costos de almacenamiento, ya que los materiales no estarían en bodega. Por otro lado, esto puede traer la incertidumbre de si los materiales llegarán en la fecha determinada, además de que la demanda del cliente es desconocida, y no puede darse una predicción al respecto. Las necesidades de producción pueden satisfacerse más adecuadamente si se conoce la demanda de los clientes por adelantado. Adicionalmente a estos problemas, es necesario considerar los riesgos que existen en el caso de que los proveedores se nieguen a vender materiales (debido a su capacidad de producción limitada y a la competencia en el mercado), o entreguen los materiales después de la fecha en la que eran requeridos.

Una estrategia de abastecimiento mixta, puede consistir de que el agente coloque una combinación de órdenes con un tiempo largo y con un tiempo corto, de tal forma que si el resto de los agentes que están compitiendo ordenan materiales con tiempos cortos, entonces el proveedor tendrá poca capacidad de producción libre, y de esa manera los precios correspondientes serán mayores que aquéllos de las órdenes con tiempo largo. En esta estrategia debe observarse la importancia de un agente para ser capaz de automáticamente cambiar su estrategia, entre el abastecimiento cercano o lejano, ya que habrá que analizar las estrategias de los otros agentes competidores.

Relacionado con la producción, también está el manejo del inventario. Una estrategia debe también considerar el nivel del inventario, tanto en materiales como en productos terminados. Por ejemplo, puede mantenerse un nivel de inventario basado en la demanda esperada del cliente por un número de días determinado, de tal forma que el límite del inventario sea calculado a partir de los productos solicitados en un determinado número de días. Otra estrategia relacionada con el inventario, sería mantener el inventario y ordenar materiales solamente después de haber ganado las órdenes de productos de los clientes, de esta forma también se reducen costos de almacenamiento, ya que tan pronto como llegan los materiales, pasan a producción, y tan pronto como son fabricados, son entregados al cliente. Una predicción adecuada de la capacidad de los proveedores y de los tiempos de retraso en las entregas, ayudaría mucho para resolver el problema. De la misma forma que con la parte de ventas, se buscarán estrategias que permitan de manera dinámica ajustar las agendas de producción, tomando en cuenta los factores ya mencionados.

5. Conclusiones y trabajo futuro

El Internet ha propiciado el cambio de los procesos de negocio individuales, hacia un modelo de negocios más distribuido, colaborativo, en un ambiente de comercio electrónico, y que se adapta a las condiciones de mercado competitivas y cambiantes. Este artículo presentó un sistema de toma de decisiones, el cual está basado en seis agentes colaborativos, explora diferentes estrategias del proceso de producción en su conjunto, y ofrece soluciones para manejar cadenas de suministro en ambientes de comercio electrónico distribuidos. El sistema está diseñado para soportar diferentes tipos de interfaces, que permitan interactuar con los modelos de negocio ya existentes, a través del manejo e integración de la cadena de suministro cooperativa.

Este sistema se encuentra en desarrollo, por lo cual el trabajo futuro incluye la implementación y pruebas de las estrategias en la plataforma de experimentación TAC SCM, en la cual se estará en posibilidad de competir con otros sistemas de agentes internacionales para medir el desempeño del sistema propuesto.

Referencias

1. Forrester, J.W.: Industrial dynamics - a major breakthrough for decision makers. *Harvard Business Review*, 36(4):37–66 (1958)
2. Cheng, F., Ettl, M., Lin, G.: Inventory-service optimization in configure-to-order systems. Technical Report RC 21781, IBM (2001)
3. Hu, J., Wellman, P.: Multiagent reinforcement learning: Theoretical framework and an algorithm. In: 15th International Conference on Machine Learning, pp. 242–250, (1998).
4. Swaminathan, J.M., Smith, S.F., Sadeh, N.M.: Modeling supply chain dynamics: A multi-agent approach. *Decision Science*, 29(30):607–632 (1998)
5. Sun, J., Sadeh, N.M.: Coordinating multi-attribute reverse auctions subjects to finite capacity considerations. Technical report, Carnegie Mellon University (2004)
6. Ghenniwa, H., Dang, J., Huhns, M., Shen, W.: Multiagent-Based Supply Chain Management, chapter eMarketPlace Model: An Architecture for Collaborative Supply Chain Management and Integration, pp. 29–62. Springer-Verlag (2006)
7. Wang, M., Liu, J., Wang, H., Cheung, W.K., Xie, X.: On-demand e-supply chain integration: A multi-agent constraint-based approach. *Expert Systems with Applications: An International Journal*, 34(4):2683–2692 (2008)
8. Jennings, N.R., Wooldridge, M.J.: *Agent Technology: Foundations, Applications, and Markets*. Springer (1998)
9. He, M., Jennings, N.R., Leung, H.: On agent-mediated electronic commerce. *IEEE Transactions on Knowledge and Data Engineering*, 15:985–1003 (2003)
10. Chaib-draa, B., Muller, J.P.: *Multiagent-Based Supply Chain Management*, volume 28 of *Studies in Computational Intelligence*. Springer-Verlag (2006)
11. Wang, Y., Fang, L.: Design of an intelligent agent-based supply chain simulation system. *Systems, Man and Cybernetics*, page 1836 1841 (2007)
12. Shena, W., Haoa, Q., Yoona, H.J., Norrie, D.H.: Applications of agent-based systems in intelligent manufacturing: *Advanced Engineering Informatics*, 20(4): 415–431 (2006)
13. Moyaux, T., Chaib-draa, B., D'Amours, S.: *Multiagent based Supply Chain Management*, chapter *Supply Chain Management and Multiagent Systems: An Overview*, pages 1–27. Springer-Verlag Berlin Heidelberg (2006)
14. Collins, J., Arunachalam, R., Sadeh, N., Eriksson, J., Finne, N., Janson, S.: The supply chain management game for the 2007 trading agent competition. Technical Report CMU-ISRI-07-100, Carnegie Mellon University (2006)
15. Trading Agent Competition. Disponible en: <http://www.sics.se/tac/>. Ultimo acceso en mayo de 2012
16. P. Pontrandolfo, P., Gosavi, A., Okogbaa, O.G., Das, T.K.: Global supply chain management: A reinforcement learning approach. *International Journal of Production Research*, 40(6):1299 1317 (2002)
17. Bapna, R., Goes, P., Gupta, A.: Predicting bidders' willingness to pay in online multiunit ascending auctions. *Inform Journal on Computing*, 20(3):345–355 (2008)
18. Skitmore, M.: Predicting the probability of winning sealed bid auctions: the effects of outliers on bidding models. *Construction Management & Economics*, 22(1):101–109 (2004)

Monitoreo de variables críticas en procesos industriales mediante una arquitectura multiagente

Luis Arturo Madrid Hurtado, Oscar Mario Rodríguez-Elias,
Germán Alonso Ruiz Domínguez y Guillermo Valencia Palomo

División de Estudios de Posgrado e Investigación, Instituto Tecnológico de Hermosillo,
Av. Tecnológico S/N, Hermosillo, Sonora, 83170, México
arturo_madrid38@hotmail.com, omrodriguez@ith.mx,
gruiz@ith.mx, gvalencia@ith.mx

Resumen Este trabajo muestra el diseño de un sistema multiagente para apoyar el monitoreo de variables críticas en procesos industriales. La propuesta ha sido validada mediante la implementación de un prototipo para monitoreo de variables de temperatura en un proceso de extrusión simulado. El sistema desarrollado se basa en una arquitectura que define un conjunto de agentes con funciones definidas, los cuales interactúan entre ellos y con diversos elementos del sistema, entre estos los dispositivos de control industrial, y bases de datos; en el caso concreto abordado en este trabajo, los dispositivos son un PLC industrial y SQL Server, una configuración común en la práctica.

Palabras clave: Sistemas multiagentes, procesos industriales, monitoreo inteligente.

1. Introducción

Los sistemas productivos están formados por un alto número de elementos de características muy dispares, que al actuar conjuntamente consiguen el objetivo del sistema: realizar productos que satisfagan las necesidades del mercado. La eficacia de estos sistemas se debe fundamentalmente a la cooperación de los elementos individuales que influyen en la eficiencia global del sistema [1], [2].

Un sistema productivo que pretenda ser competitivo en este entorno tiene que poseer ciertas características, que en otras condiciones no eran tan importantes. Aspectos como flexibilidad, adaptabilidad y rapidez de respuesta han pasado de ser aspectos deseables, a convertirse en la clave del éxito de muchas empresas.

Dentro de estos sistemas de producción existen grandes cantidades de variables y/o datos que deben ser analizados constantemente, con el fin de prevenir fallas, estancamientos o paradas en la línea de producción, para ello se utilizan sistemas informáticos o de computación que se encargan de estudiar los datos de las variables y controlarlas en forma automática.

Esto no solo ayuda a prevenir y corregir fallas dentro de los procesos de producción, sino que también podría ayudar en la optimización de dichos procesos, y lograr que el sistema se comporte de manera “inteligente”, lo que ha llevado a pensar que las nuevas tecnologías surgidas de la teoría de inteligencia artificial distribuida,

servirán para el diseño de herramientas que implementen control y supervisión inteligente a los procesos de producción.

Tomando en cuenta que el desarrollo de software basado en agentes inteligentes permite diseñar soluciones para sistemas sofisticados y complejos [3]; en este trabajo se aborda el diseño de una arquitectura multiagentes para apoyar en la captura de datos de variables críticas en un proceso industrial, como un primer paso hacia la construcción de sistemas de monitoreo y control inteligente.

2. Aplicación de los Sistemas multiagentes en Procesos industriales

Los sistemas automatizados se pueden representar mediante diferentes niveles, cada uno de los cuales tiene características operacionales adecuadas [3]: nivel de dispositivos de campo (nivel operacional), nivel de control y optimización, y nivel de gerencia de los procesos (nivel estratégico).

Los agentes inteligentes surgen como una respuesta a la necesidad de contar con aplicaciones de software que resuelvan problemas complejos minimizando la intervención externa, ya que estos son una manera de descomposición de sistemas y una alternativa razonable para implantar las funcionalidades de automatización en los diferentes niveles [4], [5]. Así, los niveles de un sistema automatizado se pueden representar por sub-sistemas, los cuales son definidos por agentes y comunidades de agentes.

El propósito de los agentes dentro de los sistemas de automatización es para proporcionar una capa adicional de acceso a la información que sea inteligente y activa, lo que permitirá una utilización más fácil y eficiente de la información para las personas que son usuarios de dicho proceso.

A continuación se muestran dos de los ámbitos principales de los sistemas de producción industrial en los cual se ha aplicado de forma más efectiva los sistemas multiagentes.

2.1. Control de procesos, monitoreo en tiempo real

Las actividades ligadas al control del proceso, el monitoreo y el diagnóstico en tiempo real están estrechamente relacionadas. El monitoreo implica la observación, el registro y el procesamiento de señales, y la detección de condiciones anormales del proceso de control. El diagnóstico es el proceso de generación de las teorías probables sobre las causas que originaron el estado actual, en este caso una anomalía dentro del sistema. En este campo se ha introducido con éxito la tecnología multiagente para un control en tiempo real flexible y distribuido [6–10].

Uno de los primeros desarrollos de esta tecnología se centro en el control de un sistema de agua helada (chiller). Para resolver el problema de la migración de tecnología, el firmware de los controladores clásicos se amplió para permitir la ejecución de agentes inteligentes directamente en el interior del controlador. Más de un agente se ejecutan en un solo controlador, y la red de controladores puede albergar una comunidad bastante grande de agentes. Esta tecnología ha sido probada con éxito

en las instalaciones de EE.UU. NAVY en Filadelfia, consiste de 116 agentes que se ejecutan en 6 Controladores [11].

2.2. Programación y el control de la producción

La programación de la producción es el proceso de selección entre planes alternativos y la asignación de recursos y tiempos a las actividades en el plan, estas asignaciones deben obedecer a un conjunto de reglas o restricciones que reflejan las relaciones temporales entre actividades, la tecnología de producción y, las limitaciones de capacidad de los recursos compartidos.

El control de fabricación se refiere a las estrategias y algoritmos para el funcionamiento de una planta de fabricación, teniendo en cuenta tanto los estados observados en el pasado como en el presente la planta de fabricación, así como la demanda del mercado. De esta manera en un sistema de fabricación basado en agentes, estos se encargan de realizar la coordinación de los recursos de producción disponibles, a fin de realizar los productos requeridos [12–15].

3. Caso de estudio: monitoreo de temperaturas críticas en un proceso de extrusión de caucho

La extrusión de materiales termoplásticos es un proceso de transformación de la materia prima, que consiste en la fusión de dicho material y una posterior inyección a través de un conducto que le dará la forma deseada. Es posible encontrar varios tipos de extrusoras, siendo la más comúnmente utilizada la extrusora mono-husillo (monotornillo).

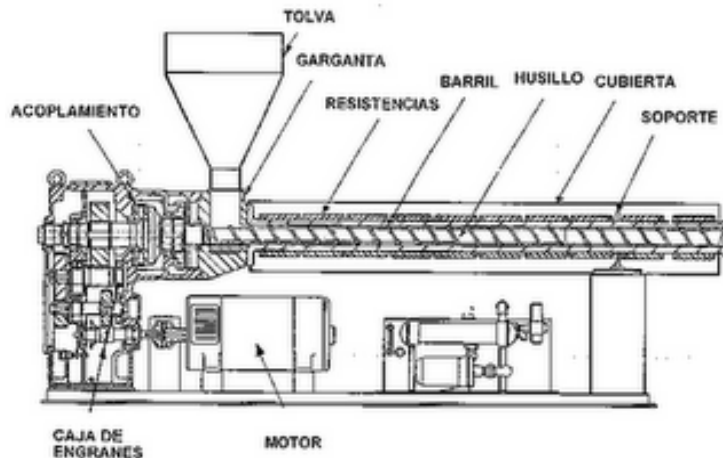


Fig. 1. Corte de Extrusor Mono-husillo (Fuente: <http://www.empaqueperformance.com.mx>).

El husillo está dividido en tres zonas bien definidas (ver Fig. 2): 1) zona de alimentación, 2) zona de compresión, 3) Zona de dosificación.

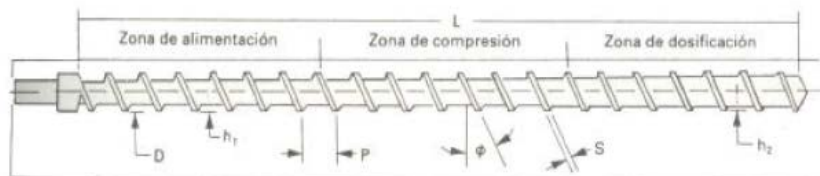


Fig. 2. Partes del husillo [16]

Dentro de este proceso uno de los factores que más incide en la calidad del producto terminado y en el aprovechamiento de la materia prima, es la temperatura durante la extrusión, por lo que el monitoreo y control de este parámetro es de vital importancia en el desempeño del proceso.

Tomando en cuenta lo anterior, en este trabajo se realizó un sistema que monitoree la temperatura de las zonas en las que está dividido el husillo del extrusor, con el fin de garantizar la estabilidad de la temperatura ya que como se mencionó anteriormente la etapa de calentamiento es una de las más críticas del proceso, debido a que el polímero debe ser fundido dentro de un rango de temperaturas específicas para que no se alteren sus propiedades.

Para esta aplicación se monitoreará el proceso de extrusión de caucho de santoprene; un material estable al calor hasta temperaturas de 246 °C [17].

Las condiciones de temperatura óptimas para la extrusión de este caucho se pueden ver en la siguiente tabla:

Table 1. Temperaturas óptimas para extrusión de caucho santoprene [17].

Zona del Extrusor	Temperatura (°C ± 5)
Zona de Alimentación.	176 °C
Zona de Compresión.	182 °C
Zona de Dosificación.	182

4. Sistema multiagente para monitoreo de temperaturas

Para apoyar en el monitoreo de variables críticas en procesos industriales, hemos diseñado una arquitectura multiagente que pueda ser adaptada a diferentes tipos de procesos. En este trabajo se ejemplifica el uso de dicha arquitectura por medio del caso de estudio ejemplo introducido en la sección anterior. A continuación se describe la forma en que se realizó el diseño del sistema siguiendo la arquitectura, así como la implementación de un prototipo para validar la viabilidad del sistema y la arquitectura propuesta.

4.1. Diseño del sistema

Esta sección muestra el diseño de un sistema basado en agentes inteligentes, que sirva como referencia para el desarrollo de sistemas de monitoreo de datos críticos

dentro de procesos de producción industrial. El sistema cuenta con una comunidad de agentes, que interactúan entre sí para llevar a cabo las tareas de supervisión de las variables más importantes dentro del proceso. Para este modelo se definieron las variables procesadas y generadas por un PLC (Controlador Lógico Programable) y los datos que son capturados por el sistema de gestión de calidad (en hojas de Excel) como las fuentes principales de datos, sobre las cuales actuarán los agentes, esto debido a que es una configuración comúnmente usada en la práctica.

La comunidad de agentes está formada por cuatro tipos de agentes, Agentes Coordinadores, Agentes de proceso, Agente Concentrador y Agentes que monitorean los datos del sistema de calidad, descritos a continuación.

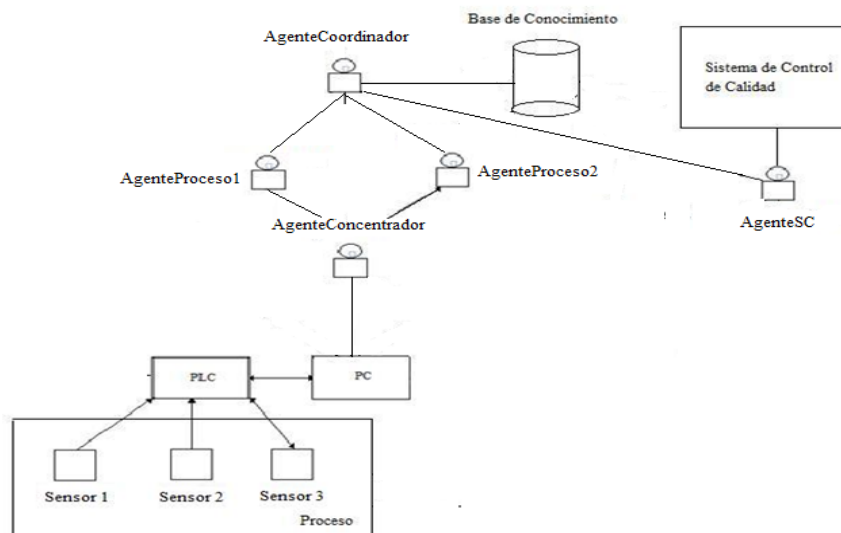


Fig. 3. Arquitectura del sistema.

- **Agente Coordinador.** Se encarga de coordinar la información que será requerida del proceso, solicitándola a los agentes de proceso. También se encarga de recopilar y almacenar dicha información en una base de datos.
- **Agente de Proceso.** Se encarga de monitorear una o varias variables específicas del proceso, y de enviarlas al agente coordinador cuando éste se la solicite o cuando ocurra un cambio significativo en las variables.
- **Agente Concentrador.** Establece el enlace con el PLC, para obtener el valor de las variables importantes del sistema, y mandársela a los agentes de procesos cuando estos las soliciten.
- **Agente Monitoreador De SGC.** Monitorea los datos capturados para el sistema de calidad, estableciendo el enlace con las hojas de cálculo donde son capturados, para obtener su valor y mandarlos al agente coordinador cuando este lo solicite o cuando ocurran cambios significativos.

La figura 3 muestra la arquitectura del sistema. Es posible observar las interacciones que se dan entre los diferentes tipos de agentes, y donde el objetivo

general del sistema es almacenar las variables más importantes en una base de conocimiento para su posterior análisis.

4.2. Implementación del prototipo

La implementación de los Agentes se hizo sobre la plataforma JADE (<http://jade.tilab.com/>), como se muestra en la figura 4. El desarrollo de las aplicaciones se realizó en NetBeans 7.0, con java JDK 1.6.

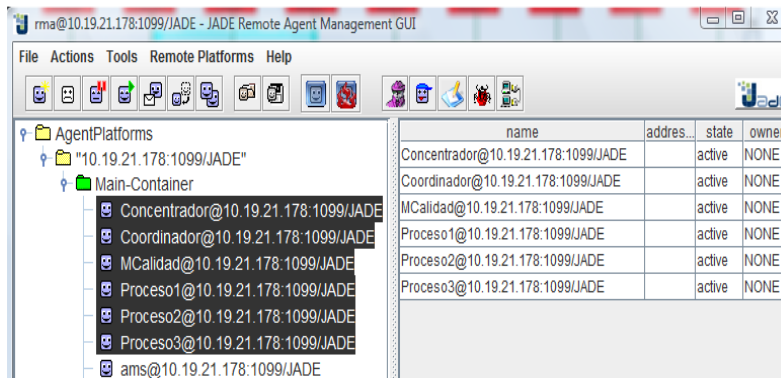


Fig. 4. Implementación de agentes en JADE.



Fig. 5. Simulación de temperaturas de Extrusión en LabView.

Para implementar el sistema fue necesario contar con una aplicación que le enviara datos al PLC, simulando el proceso de producción. Para esto se utilizó LabView, que permitió simular las variaciones de temperaturas en las diferentes zonas del extrusor que los agentes estaban encargados de monitorear (ver figura 5).

Con ayuda del Agente utilitario Sniffer de JADE, como se muestra en la figura 6, es posible observar los mensajes enviados por los agentes de la comunidad, esto con

el fin de cumplir sus objetivos. Para la comunicación de los agentes fue necesario generar una ontología que manejara los conceptos básicos del lenguaje de comunicación.

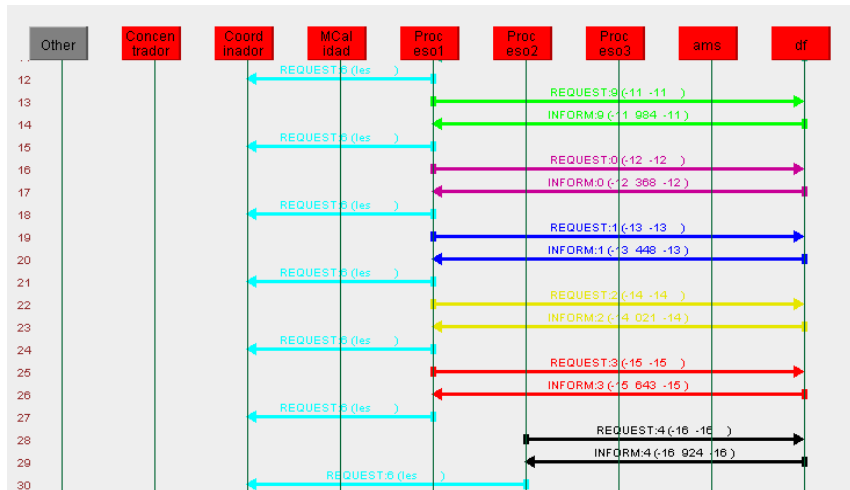


Fig. 6. Mensajes entre Agentes en caso de estudio.

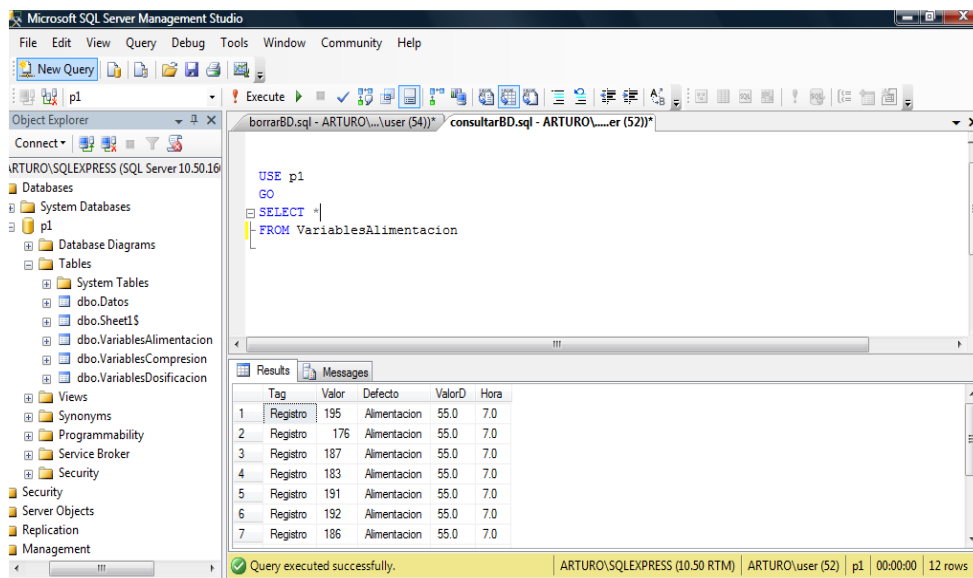


Fig. 7. Almacenamiento de cambios significativos en base de datos.

Finalmente en la figura 7 se muestra como los agentes almacenan, en una base de datos en Microsoft SQL server 2008, los cambios significativos de las variables, en este caso un incremento de 10 °C en las temperaturas ideales de extrusión del caucho de santoprene.

5. Conclusiones y comentarios finales

En este trabajo se ha mostrado el diseño de una arquitectura multiagente genérica para apoyar en el monitoreo de variables críticas en procesos industriales. La arquitectura ha sido validada con la implementación de un prototipo para el monitoreo de variables críticas de temperatura en un proceso de extrusión simulado. Se pudo comprobar que los agentes desarrollados para este caso de estudio, cumplieron con las actividades de monitoreo y almacenamiento de los cambios significativos en las variables críticas del sistema, lo cual es el primer paso, a fin de que estas variables sean posteriormente analizadas y sirvan para una adecuada toma de decisiones.

Cabe destacar que para la implementación del sistema se han aprovechado las capacidades de transmisión de datos vía Ethernet de los dispositivos de control industrial, en nuestro caso un PLC con un módulo web. El siguiente paso en la implementación del sistema multiagente es agregar las fases de análisis de los datos capturados, para posteriormente apoyar en la mejora del sistema de control de forma inteligente, flexible y en tiempo real, lo cual constituye nuestro trabajo futuro.

El objetivo de este trabajo en el mediano y largo plazo, es proveer herramientas que apoyen en la mejora de los sistemas de automatización y control industrial, integrándolos con los sistemas de manejo de información y toma de decisiones mediante el uso de tecnologías de inteligencia artificial, particularmente agentes de software, que apoyen en la gestión de datos, información y conocimiento al interior de los procesos de producción, vía la comunicación directa con los dispositivos de control industrial.

Agradecimientos. Este trabajo ha sido financiado parcialmente por PROMEP (oficio 103.5/11/840). Se agradece el apoyo de Conacyt con la beca numero 58920.

Referencias

1. M. P. Groover: Automation, Production Systems, and Computer Integrated Manufacturing. 3rd ed. Prentice Hall, p. 840 (2007)
2. J. A. Araúzo, J. J. De Benito Martín, R. del Olmo Martínez, and P. A. Angulo Sanz: Situación actual y expectativas de los sistemas de fabricación basados en agentes. In: VIII Congreso de Ingeniería de Organización, pp. 1043-1052 (2004)
3. A. Ríos-Bolívar, M. Cerrada, F. Narciso, F. Hidrobo, and J. Aguilar: Implantando sistemas de control con agentes inteligentes. Revista Ciencia e Ingeniería, vol. 29, no. 3, pp. 249-260 (2008)
4. C. Bravo, J. Aguilar, and F. Rivas: Diseño de una arquitectura de automatización industrial basada en sistemas multiagentes. Revista Ciencia e Ingeniería, vol. 25, no. 2, pp. 75-88 (2004)
5. T. Wagner: An agent-oriented approach to industrial automation systems. Agent Technologies, Infrastructures, Tools, and Applications for E-Services (LNCS), vol. 2592, pp. 314-328 (2003)
6. W. Shen, D. H. Norrie, and R. Kremer: Developing Intelligent Manufacturing Systems Using Collaborative Agents. In: 2nd International Workshop on Intelligent Manufacturing Systems, pp. 22-24 (1999)
7. V. Julian and V. Botti: Developing real-time multi-agent systems. Integrated Computer-Aided Engineering, vol. 11, no. 2, pp. 135-149 (2004)

8. J. H. Taylor and A. F. Sayda: Intelligent Information, Monitoring, and Control Technology of Industrial Process Applications. In: 15th International Conference on Flexible Automation and Intelligent Manufacturing (2005)
9. C. R. Durán Fernández: Control Distribuido Auto-organizado para un Sistema Flexible de Manufactura. Universidad del Bío-Bío (2005)
10. L. Bunch et al.: Software agents for process monitoring and notification. In: Proceedings of the 2004 ACM symposium on Applied Computing - SAC'04, pp. 94-99 (2004)
11. M. Pechoucek and V. Marik: Industrial deployment of multi-agent technologies: review and selected case studies. *Autonomous Agents and Multi-agent Systems*, vol. 17, no. 3, pp. 297-431 (2008)
12. M. Cerrada, J. Aguilar, J. Cardillo, and R. Faneite: Agent-based design for fault management system in industrial processes. *Revista Técnica de Ingeniería de la Universidad del Zulia*, vol. 29, no. 3, pp. 258-268 (2006)
13. P. Leitão and F. Restivo: A Framework for Distributed Manufacturing Applications. In: *Advanced Summer Institute Life Cycle Approaches to Production Systems: Management, Control and Supervision*, pp. 75-80 (2000)
14. R. M. Lima, R. M. Sousa, and P. J. Martins: Distributed production planning and control agent-based system. *International Journal of Production Research*, vol. 44, no. 18-19, pp. 3693-3709 (2006)
15. V. Marík and D. McFarlane: Industrial Adoption of Agent-Based Technologies. *IEEE Intelligent Systems*, vol. 20, no. 1, pp. 27-35 (2005)
16. M. De Antonio Gómez: Control de un Extrusor de Perfiles de ETP. (2004)
17. Avanced Elastomer Systems, *Boletín General: El Caucho vulcanizado que se procesa como termoplástico*. (2002)

Lógica y razonamiento

Complejidad computacional de un caso especial del cálculo de grupos en las redes de interacción

Roberto Alonso, Raúl Monroy y Luis A. Trejo

Departamento de Ciencias Computacionales
Instituto Tecnológico y de Estudios Superiores de Monterrey,
Campus Estado de México, México
{A01013374,raulm,ltrejo}@itesm.mx

Resumen En este trabajo proponemos una aproximación para realizar una demostración de la complejidad computacional del problema del Cálculo de Grupos. El Cálculo de Grupos es un problema del modelo social conocido como Redes de Interacción[5] que resulta ser complejo. Mediante reducciones de Karp realizamos una demostración de un caso especial de nuestro problema que consideramos es esencial para lograr una demostración completa del problema del Cálculo de Grupos. Si se determina la complejidad computacional del Cálculo de Grupos, nos aseguramos de que el problema es realmente complejo y sólo puede ser resuelto parcialmente mediante heurísticas.

1. Introducción

En el modelo Redes de Interacción [5] existe un problema llamado *problema del Cálculo de Grupos*, este cálculo de estructuras sociales es necesario para entender mejor el comportamiento de un grupo de usuarios que hacen uso de recursos. Este conocimiento a su vez será útil para la detección de ataques de Denegación de Servicio Distribuidos (DDoS por sus siglas en inglés) dirigidos a servidores DNS.

Para entender mejor los problemas, la teoría de la complejidad clasifica los problemas en función de su dificultad para resolver estos. Típicamente la clase P representa aquellos problemas que pueden ser resueltos en tiempo polinomial, mientras que la clase NP representa aquellos problemas que sólo pueden ser resueltos con una máquina no-determinística de Turing (NDTM por sus siglas en inglés). En este trabajo mostramos que un caso especial del Cálculo de Grupos (**SGC**) es un problema NP-Completo. El paso presentado en este trabajo será esencial para demostrar que la versión generalizada del problema es también NP-Completo.

Para demostrar que el Cálculo de Grupos (SGC) en su forma generalizada es NP-Completo requerimos de hacer una reducción en tiempo polinomial de un problema NP-Completo a nuestro problema. Además es necesario demostrar que una respuesta en el problema NP-Completo existe *si* hay una respuesta en nuestro problema.

El trabajo está dividido de la siguiente manera: En la sección Redes de Interacción y Complejidad del cálculo de grupos abordamos el tema del modelo social y mostramos el cálculo de la complejidad temporal para calcular todas las posibles estructuras sociales. De forma general presentamos la reducción de Karp, en la sección 3, que consiste en una técnica para realizar de forma simple demostraciones de problemas nuevos. En la sección 4 mostramos cómo se realizó la reducción del problema del *Hitting Set* al caso especial del SGC. En las conclusiones, en la sección 5, presentamos algunas ideas de como puede usarse la reducción de este trabajo para demostrar que el problema general de decisión del SGC es NP-Completo.

2. Redes de interacción y complejidad del cálculo de grupos

La hipótesis manejada en las Redes de Interacción, se basa en el uso de perfiles sociales derivados de la caracterización de usuarios de red que hacen uso del servicio DNS, y en el cual se observan las interacciones que tienen estos usuarios con un servidor DNS. Un **usuario** es aquella entidad en una red que solicita resolución DNS. Por lo tanto, consideraremos que todas las peticiones realizadas a través de un dispositivo de red, sin importar si es realizada por un software o un usuario de computadora, son peticiones de un mismo **usuario** si y sólo si, provienen de la misma IP.

El objetivo principal de este enfoque, es el de caracterizar los perfiles sociales de los usuarios, ya que esto nos permitirá observar comportamientos de uso de recursos DNS. Es decir, estaremos definiendo características de los usuarios que nos permitirán distinguir lo que denominaremos como normalidad social y anormalidad social. Por comodidad este modelo es nombrado **Redes de Interacción** y será representado como un grafo nombrado *grafo de conectividad*.

Al obtener parte de la caracterización social, podemos esperar relaciones entre los dominios que visita un usuario y otro. Esto quiere decir que nuestros hábitos de navegación a través de Internet son similares o iguales a los de otra persona. Por ejemplo, el dominio `www.ask.com` sería una de las últimas opciones como buscador a comparación de `www.google.com`. Por esa razón, sabemos que existen patrones de consultas comunes realizadas a un servidor DNS recursivo, estas consultas comunes entre más de un usuario se denomina **grupo**.

2.1. Complejidad temporal del cálculo de grupos

Un **grupo** es un conjunto de k usuarios que tienen en común la consulta de n recursos. La complejidad temporal para determinar todos los posibles grupos en una Red de Interacción está dada por la siguiente fórmula:

$$\sum_{k=2}^{N \leq r} C(r, k)$$

donde r es el número de recursos únicos en un conjunto de peticiones, y N es el tamaño del grupo más grande asumiendo que conocemos ese tamaño o que el algoritmo continúa corriendo hasta que no encuentre un grupo.

Encontrar todos los grupos posibles dado un *grafo de conectividad* es un problema intratable y lo llamaremos problema **SGC general**. El problema generalizado del SGC requiere de grandes recursos computacionales y puede ser considerado como un problema de enumeración en el que calcular y escribir la respuesta es igual de complejo para ciertas instancias.

Desde el punto de vista de la teoría de la complejidad computacional, debemos acotar la versión del SGC general a una versión de decisión que pueda ser resuelta con una NDTM, esta acotación será nombrada **SGC- δ** . Los problemas de decisión son aquellos que pueden ser contestados con la respuesta 'sí' en caso de existir una respuesta válida y 'no' en caso de que sea inexistente una respuesta válida para el problema. Demostrar que la versión de decisión del problema del Cálculo de Grupos es uno de los pasos para comprobar que el problema **SGC general** es complejo.

En este trabajo mostramos que un caso especial del Cálculo de Grupos es NP-Completo. Esta demostración podría ser usada para demostrar que **SGC- δ** es NP-Completo y posteriormente que el problema **SGC general** está en la clase #P.

3. Determinar la complejidad computacional de problemas poco conocidos

La teoría de la complejidad clasifica los problemas en función de la dificultad para resolverlos. Asumiendo que $P \neq NP$, los problemas en P son aquellos tratables mientras que los NP son la clase de problemas que sólo pueden ser resueltos por una Máquina de Turing No-Determinista.

El problema del cálculo de grupos no pertenece a la clase P dado que la cantidad de recursos computacionales necesarios para resolverlo crece exponencialmente con respecto al número de dominios consultados por los usuarios. Sin embargo, es insuficiente decir que el problema es NP-Completo sólo porque se sospecha no pertenece a P por tanto, usaremos una Reducción de Karp [1,2] para demostrar que el problema del Cálculo de Grupos es NP-Completo.

Una Reducción de Karp es una de las formas para demostrar que un problema pertenece a la clase NP. Está basada en el trabajo que realizó Cook-Levin acerca de la primera demostración de un problema NP. Karp establece por transitividad que si un problema cuya clase es desconocida puede ser mapeado de un problema NP-Completo como SAT, entonces hay un prueba de que el problema desconocido (π) pertenece a la clase NP. Es decir,

$$NP \leq_p SAT \leq_p \pi$$

por transitividad,

$$NP \leq_p \pi$$

Una reducción de Karp es uno de los métodos preferidos para demostrar si un nuevo problema es NP-Completo. Para realizar una reducción de Karp se deben seguir los siguientes pasos:

- Demuestra que el problema está en NP.
- Selecciona un problema NP-Completo.
- Realiza una reducción en tiempo polinomial del problema NP-Completo al problema a ser demostrado.

Una reducción simplemente se refiere al mapeo de un problema a otro problema. La reducción siempre se tiene que hacer del problema conocido NP al problema No-conocido.

4. Reduciendo el problema de Hitting Set a un caso especial del problema del cálculo de grupos

La versión generalizada del problema del Cálculo de Grupos (**SGC**) pretende determinar todos los posibles grupos dado un grafo de conectividad. Sin embargo, para realizar la demostración de que un problema es NP-Completo, requerimos de hacer una acotación de tal forma que sea posible definir **SGC** como un problema de decisión.

4.1. Problema de decisión del SGC y el problema del Hitting Set (HS)

Por comodidad, la versión de decisión del problema **SGC**, será nombrada **SGC- δ** y está definida de la siguiente forma:

Dados un grafo de conectividad $G = (V, E)$, dos conjuntos disjuntos $I = \{i_1, i_2, i_3, \dots, i_k\}$ y $D = \{d_1, d_2, d_3, \dots, d_l\}$ representando los nodos $V = I \cup D$ y las consultas de los usuarios i_k representada por los arcos, ¿existe un grupo de tamaño 2?

El problema anterior es la definición del problema **SGC- δ** que es de interés para las Redes de Interacción. Sin embargo, vamos a utilizar una aproximación similar a la presentada en [3,4] para hacer la demostración de ese problema. Dicha aproximación consiste en demostrar un caso especial del problema general **SGC** para utilizarla en el problema de interés. Para lograr lo anterior definimos un nuevo problema **SGC** nombrado **SGC- σ** de la siguiente forma:

Dado un grafo de conectividad $G = (V, E)$, ¿existe una interacción completa? Es decir, en el grafo de conectividad, ¿todos los usuarios forman parte de al menos un grupo de tamaño dos ?

Proponemos realizar una reducción de Karp utilizando el problema de Hitting Set a un caso particular del problema del Cálculo de Grupos (**SGC**). Hitting Set (**HS**) es un problema NP-Completo definido de la siguiente forma:

Dado un conjunto finito S , una colección C de subconjuntos de S , un número positivo $K \leq |S|$, ¿existe un subconjunto $S' \subseteq S$ tal que S' contiene al menos un elemento de cada subconjunto de C y $|S'| \leq K$?

Teorema 1 (SGC- σ es NP completo).

Demostración. El problema está en NP porque una NTDM puede resolver el problema en tiempo no-determinístico y verificar la respuesta en tiempo polinomial. Para hacer la transformación $\mathbf{HS} \leq_p \mathbf{SGC}\text{-}\sigma$, debemos de construir un grafo de conectividad $G(V,E)$ tal que existe una interacción completa si y sólo si existe un HS en S.

Podemos transformar \mathbf{HS} a $\mathbf{SGC}\text{-}\sigma$ de la siguiente forma. Sea $S = \{s_1, s_2, s_3, \dots, s_n\}$, sea $C = \{C_1, C_2, C_3, \dots, C_m\}$ una colección de subconjuntos de S, y sea $K \leq |S|$ una instancia de \mathbf{HS}

Por cada elemento s_i vamos a construir un componente nombrado subgrafo social. El subgrafo social se construye de la siguiente forma:

1. Agrega un usuario y dos recursos que estén siendo consultados por este usuario. (Fig. 1).

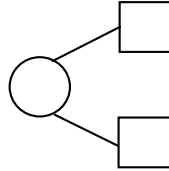


Figura 1. Usuario consultando dos recursos.

2. Si existe una colección C_j que contenga s_i agregar un usuario etiquetado como C_j conectado a los recursos agregados recientemente. (Fig. 2).

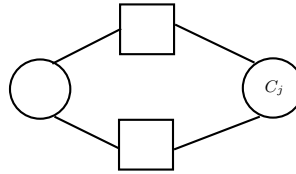


Figura 2. Usuario C_j conectado a dos recursos.

3. Agregar un usuario conectado a uno de los recursos que no tenga grado mayor a 2 y agregar un recurso más conectado al nuevo usuario. (Fig. 3).
4. En caso de existir otro C_j que contenga s_i repetir los pasos 2 y 3.
5. Si ya no existen más C_j que contengan s_i encerrar el componente en una caja etiquetada s_i . (Fig. 4).

Cada colección C_j corresponde a un usuario i_j en el problema $\mathbf{SGC}\text{-}\sigma$. Por simplicidad, el usuario está representado como un círculo y etiquetado en el grafo que se construye como *usuario* C_j , este usuario hace consultas a recursos 'dummy'. Dado que nos interesa la formación de una interacción completa, no es relevante con que recursos se esté formando esa relación.

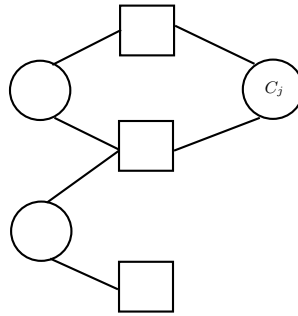


Figura 3. Usuario nuevo encadenado al componente.

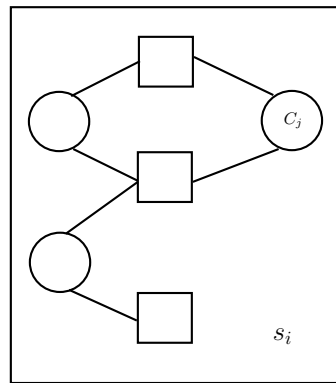


Figura 4. Usuario nuevo encadenado al componente.

Ejemplo de transformación dada una instancia del HS. Sea $S = \{s_1, s_2, s_3, s_4\}$ y $C_1 = \{s_1, s_2, s_3\}$, $C_2 = \{s_2, s_4\}$, $C_3 = \{s_1, s_2, s_4\}$ y $K=2$ una instancia del problema HS. Siguiendo la transformación obtenemos un grafo que puede contener una interacción completa. (Fig. 5).

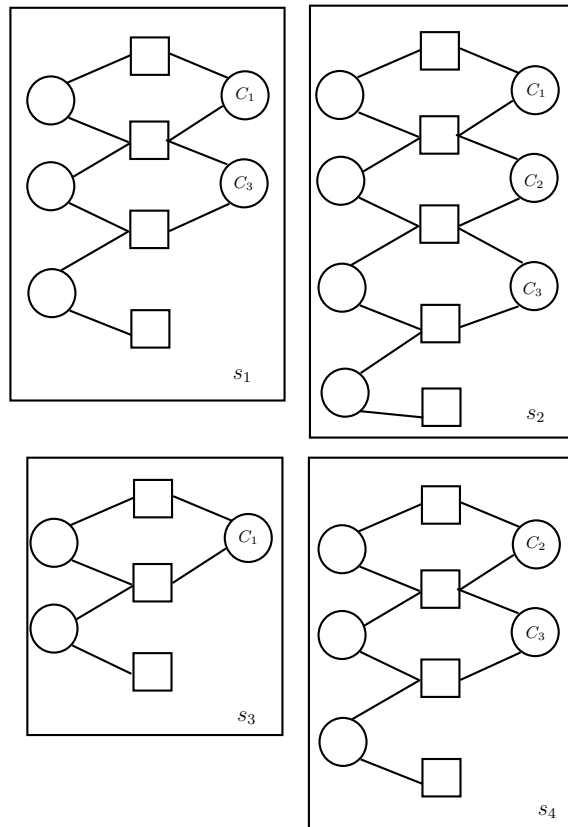


Figura 5. Grafo resultante de la instancia $S = \{s_1, s_2, s_3, s_4\}$ y $C_1 = \{s_1, s_2, s_3\}$, $C_2 = \{s_2, s_4\}$, $C_3 = \{s_1, s_2, s_4\}$.

Una vez transformado el problema, es necesario demostrar que existe una solución en el problema NP-Completo si y sólo si existe una solución en el problema nuevo. Es decir, si existe un HS en la instancia presentada, debe existir una interacción completa en el grafo y si no hay una respuesta válida en la instancia de HS tampoco se debe encontrar una interacción completa en el grafo. Recordando el ejemplo anterior, una respuesta del problema HS válida sería $S' = \{s_1, s_4\}$ ya que contiene al menos un elemento de cada C_j y además $|S'| = 2$.

Para determinar si es posible encontrar una interacción completa seleccionamos los componentes correspondientes en el grafo. La unión de ambos componentes $\{s_1\} \cup \{s_2\}$ es el grafo que debe contener la interacción completa (Fig. 6).

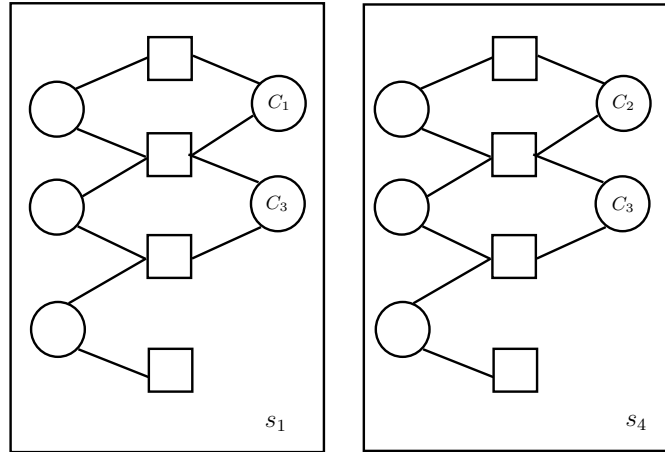


Figura 6. Resultado de la unión de s_1 y s_4 .

Podemos notar que al seleccionar esos dos componentes, estamos encontrando que todos los usuarios intervienen en la formación de al menos un grupo de tamaño dos por lo que es una respuesta válida en el problema **SGC- σ** .

Si seleccionamos $S' = \{s_1, s_3\}$ como respuesta del problema HS podemos notar que *no* es una respuesta válida. Seleccionando los componentes correspondientes en el grafo de conectividad (Fig. 7), podemos notar que no todos los usuarios están formando al menos un grupo.

Por lo tanto, queda demostrado que $HS \leq_p \mathbf{SGC-}\sigma$ y en consecuencia $NP \leq_p \mathbf{SGC-}\sigma$. ■

La importancia de la reducción en el caso especial radica en que tenemos un problema NP-Completo comprobado que está dentro del dominio de nuestro problema y por tanto, puede ser utilizado para realizar la demostración de $SGC-\delta$. Es decir, $\mathbf{SGC-}\sigma \leq_p \mathbf{SGC-}\delta$ es una aproximación para demostrar que encontrar un grupo de tamaño dos es complejo.

5. Conclusiones y trabajo futuro

Las redes de interacción es un modelo temporal que se construye a partir del análisis de datos sociales. Está basado en la caracterización de los usuarios y los recursos. Los usuarios y los recursos tienden a formar grupos con respecto a las preferencias que tienen estos usuario. Calcular estos grupos es esencial para la

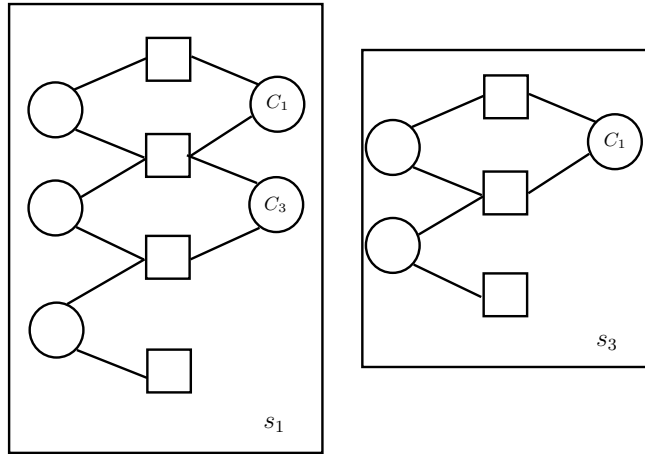


Figura 7. Resultado de la unión de s_1 y s_3 .

detección de anomalías sociales y en última instancia la detección temprana de ataques DDoS a servidores DNS.

El cálculo de grupos es complejo dado que no existe un algoritmo polinomial capaz de resolver el problema de forma correcta y completa para cualquier instancia del problema. Por tanto, se requiere de diferentes métodos para aproximar una solución completa. Podemos aplicar diferentes métodos para aproximar el cálculo de estos grupos, algunos relacionados con segmentación de imágenes o manipulación de matrices.

Queda demostrado que el caso especial del problema SGC está en la clase NP y es NP-Completo. Esta demostración puede ser usada para continuar la reducción al problema SGC- δ ya que ambos problemas se encuentran en el mismo dominio. Podemos utilizar la reducción SGC- σ simplemente copiando la instancia del problema. Si existe una interacción completa es evidente que existe un grupo de tamaño 2, aunque si existe un grupo de tamaño 2 no implica la existencia de una interacción completa.

Los resultados obtenidos en este trabajo son prometedores. A pesar de no ser el problema que buscamos exactamente es una aproximación y un paso inicial para demostrar que la versión SGC- δ es NP-Completo. Más aún, puede darnos una idea de cómo hacer la demostración de la versión general del SGC en el que se puede preguntar si existen varios grupos de diferentes tamaños, lo cual se sospecha también es un problema complejo.

Al demostrar la versión generalizada del SGC podemos concluir que la versión de optimización de nuestro problema es NP-Hard y la versión de enumeración pertenece a la clase #P. Este conocimiento nos ayudará a entender mejor el problema del Cálculo de Grupos y proponer diferentes heurísticas que nos ayuden a resolver el problema de manera eficiente aunque incompleta.

Referencias

1. Karp, R.M.: Reducibility Among Combinatorial Problems, Complexity of Computer Computations, Miller, R.E and Thatcher, J.W. editors, pp. 85-103 (1972)
2. Garey M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman and Company (1991)
3. Papadimitriou, Ch.: The NP-Completeness of the bandwidth minimization problem, Springer Wien, pp. 263-270 (1976)
4. Blanche, G., Karpinski, M., Wirtgen, J.: On Approximation Intractability of the Bandwidth Problem (1997)
5. Alonso, R., Vázquez, J., Trejo, L., Monroy, R., Sánchez, E.: How Social Networks can Help to Detect DDoS Attacks on DNS Servers, Proceedings of the 3rd MICAI Workshop on Computer Security in 8th Mexican International Conference on Artificial Intelligence, WSEC09-MICAI '09 (2009)

Lenguaje P-log: estructura y traducción a SMOBELS

Daniel Rodríguez, Claudia Zepeda y José Luis Carballido

Benemérita Universidad Autónoma de Puebla (BUAP),
Facultad de Ciencias de la Computación, Puebla, México
rmdaniel.04@gmail.com, czepedac@gmail.com, jlcarballido7@gmail.com

Resumen En este artículo estudiaremos el lenguaje P-log, el cual posee una base lógica y una base probabilística. P-log es un lenguaje que sirve para modelar y resolver problemas de probabilidad basándose en la programación lógica. P-log tiene una base lógica que es soportado por *Answer Set Programming* y la base probabilística por Redes bayesianas. *Answer Set Programming* tiene diferentes implementaciones y P-log utiliza SMOBELS. Nuestro artículo tiene como propósito analizar el lenguaje P-log y mostrar como se realiza una traducción a una implementación del lenguaje ASP, es decir, traducirlo a SMOBELS.

Palabras clave: P-log, *answer set programming*, SMOBELS.

1. Introducción

La programación lógica es un tipo de programación declarativa que se forma por un conjunto de reglas. Un programa lógico para que pueda ser entendido se le debe asignar una semántica. La semántica es la manera de determinar el tipo de conclusiones que se pueden establecer a partir del conjunto de reglas, una de éstas semánticas es la semántica estable o semántica de modelos estables (SE) [1, 6]. La SE permite modelar el concepto de negación como falla en los programas lógicos y es la base de los *Answer Set Programming* (ASP).

ASP es un lenguaje de representación de conocimiento basado en SE de programas lógicos con negación como falla [2-3]. En ASP los programas se reducen a calcular los modelos estables. ASP tiene diferentes implementaciones como: POTASSCO¹, DLV², SMOBELS³, etc., y diferentes aplicaciones como planificación [7], actualización, razonamiento sobre acciones y causalidad, modelado de problemas con probabilidad [4].

Entre las aplicaciones que hemos mencionado de ASP, se encuentra el modelado de problemas probabilísticos, y uno de los lenguajes desarrollados para modelar este problema es P-log [4]. P-log es un lenguaje declarativo que posee una base lógica y una base probabilística. La base lógica es soportada por ASP y la base probabilística por las redes bayesianas. Debido a que P-log está basado en ASP, debemos realizar una transformación de un programa en lenguaje P-log al lenguaje ASP y obtener los modelos estables gracias a una implementación de ASP que es SMOBELS. Los mo-

¹ <http://potassco.sourceforge.net/>

² <http://www.dlvsystem.com/>

³ <http://www.tcs.hut.fi/Software/smodels/>

delos estables que están asociados al programa ASP corresponden a los mundos posibles asociados a P-log, además P-log posee otra característica muy importante que es la de agregar información probabilística a los mundos posibles. Esta información probabilística es calculada por P-log, sin embargo, en este artículo no nos enfocaremos en saber cómo los asigna. La finalidad de P-log es saber que probabilidad hay de que ocurra un suceso, y dependiendo de esas probabilidades tomar decisiones. Para obtener las probabilidades en el programa P-log, se realiza una consulta al final del programa.

El propósito de este artículo es presentar de forma muy detallada y sencilla la traducción de un programa escrito en lenguaje P-log a un programa que entienda SMOBELS. En estos tiempos, aún no existe un manual o artículo que hable de la traducción de un programa en P-log a SMOBELS, nosotros estamos presentando esta traducción para entender como trabaja la semántica en P-log. Cómo habíamos dicho P-log modela y resuelve problemas de probabilidad, que pueden ser de ayuda al momento de tomar decisiones tomando en cuenta probabilidades.

En la sección 2 veremos algunos conceptos que utilizaremos durante este artículo. En la sección 3 mostraremos como se traduce un programa de lenguaje P-log a SMOBELS. En la sección 4 daremos algunas conclusiones de este trabajo.

2. Preliminares

En esta sección vamos a familiarizarnos con algunos conceptos con los cuales trabajaremos durante este artículo.

2.1. Answer Set Programming

ASP es un lenguaje de representación de conocimiento y fue introducido alrededor de 1990. ASP está basado en la semántica de modelo estable de programas lógicos con negación como falla [4].

Un programa en ASP corresponde a un conjunto de reglas, donde las reglas son de la forma: $L_0 \leftarrow L_1, \dots, L_m, \text{not } L_{m+1}, \dots, \text{not } L_n$, donde L_i s son literales⁴, $m \geq 0$, y $n \geq m$, en programación lógica cambiamos " \leftarrow " por el símbolo " $:-$ ". Las partes de la izquierda y derecha de " \leftarrow " se llaman la cabeza y el cuerpo de la regla, respectivamente. Cuando el cuerpo de la regla es vacío y en la cabeza hay una sola literal se llama un hecho, este se puede escribir sin el símbolo " \leftarrow " por ejemplo: " L_0 ". Cuando la cabeza es vacía, la regla se conoce como restricción, por ejemplo " $L_1, \dots, L_m, \text{not } L_{m+1}, \dots, \text{not } L_n$ ".

SMOBELS, es un implementación de ASP que trabaja con SE para programas lógicos con negación como falla [5]. Los programas hechos en SMOBELS son escritos usando notaciones de programación lógica básica, es decir, programas compuestos de átomos y reglas de inferencia.

⁴ Literales son átomos o átomos negados (precedidos del símbolo " $-$ ").

2.2. Lenguaje P-log

Un programa en P-log se conforma de las siguientes partes: declaraciones, reglas de selección random, información probabilística, observaciones y acciones. La manera más detallada de cada una de estas partes la encuentra en [4]. Vamos a describir cada una de las partes de manera breve y para que sea más fácil de entender a los lectores mostraremos un ejemplo sencillo. *El problema que se ilustra como ejemplo trata de 4 corredores que están compitiendo en las olimpiadas, y se asigna un juez de línea que está en la salida, éste indica si la salida fue buena o fue salida en falso, entonces habrá un ganador si hay una salida buena. La finalidad de este programa es saber que probabilidad hay de que un corredor llegue en primer lugar. Podemos decir que el corredor 2 tiene una probabilidad de ganar del 20%(0.20).*

Declaraciones: La declaración en un programa P-log puede tener una o todas de las siguientes formas: *definición de dominio de tipo, definición de tipos, declaración de tipos para atributos, declaración de la parte regular.*

Una *definición de dominio de tipo* en P-log tiene el siguiente formato:

```
#domain Tipo(Variable).
```

donde `Tipo` es un identificador para nombrar el tipo, el identificador debe ir con letras minúsculas. `Variable` es un identificador que puede ser una palabra que inicie con letra mayúscula, o simplemente, una letra mayúscula.

Tabla 1. En nuestro problema tenemos 2 dominios, uno es `corredor`, para designar a los corredores, y otro es `boolean` que designa al juez de línea.

#domain corredor(C).
#domain boolean(B).

La *definición de tipos* en P-log puede ser de 2 formas:

- en forma de lista y tiene el siguiente formato: $\text{Tipo} = \{X_1, \dots, X_n\}$.
- en forma de rango y tiene el siguiente formato $\text{Tipo} = \{X_1 \dots X_n\}$.

donde `Tipo` es un identificador con letras minúsculas, o bien, que pudo haber sido definido como dominio de tipo o ser definir como uno nuevo. X_i , con $1 \leq i \leq n$, es una lista de valores.

Tabla 2. Definimos los 2 tipos que habíamos declarado en la tabla 1, observamos que tenemos 4 corredores y 2 valores de `boolean` que indican si la salida es buena o falsa, cabe mencionar que se puede definir tipos sin declararlo en un dominio.

corredor = {1 .. 4}.
boolean = {b, f}.

La *declaración de tipos para atributos* en P-log puede ser de 2 formas:

- Atributo con dominio y rango, tiene el siguiente formato:
`Atributo: Dominio -> Rango.`
- Atributo solo con rango, tiene el siguiente formato: `Atributo: Rango.`

donde `Dominio`, `Rango` son definiciones de tipo. `Atributo` es un identificador que debe ser escrita en letras minúsculas.

Tabla 3. Declaramos un atributo `primero` de tipo `corredor`, un atributo `salida` de tipo `boolean` y un atributo `ganador` que mapea un tipo `corredor` a un tipo `boolean`, `primero` y `boolean` tendrán los valores que se les asignaron a los tipos.

<code>primero : corredor.</code>
<code>salida : boolean.</code>
<code>ganador : corredor -> boolean.</code>

La *declaración de la parte regular* de un programa P-log consiste en una colección de reglas formadas usando literales. Éstas reglas pueden ir o no en el programa.

Tabla 4. En esta parte se define que si la `salida` fue falsa no habrá ningún `ganador`, y si no hay evidencia de que haya una `salida` en falso, es decir, que la `salida` es buena, entonces habrá un `ganador` de la carrera.

<code>ganador(C) = b :- not ganador(C) = f.</code>
<code>ganador(C) = f :- salida = f.</code>

Reglas de selección random. Existen 2 formas de declarar reglas random en P-log: *declarando reglas random con un atributo y una función de condición*, *declarando reglas random un atributo y además agregando cuerpo a la regla*.

Las *reglas random con un solo atributo y una función de condición* tienen el siguiente formato:

`[Nombre_regla] random(Atributo:{X:Función(X)}).`

donde `Nombre_regla` es un identificador para la regla, `Atributo` fue declarado en la sección de declaraciones. `Función(X)` es una regla que debe cumplirse para que el valor obtenido de la selección aleatoria sea verdadero.

Tabla 5. En esta regla se define que se escogerá un valor aleatorio para el atributo `primero`, pero debe cumplir que la función que sea verdadera, y para eso se utilizan las reglas escritas en la tabla 4.

<code>[r1] random(primero:{X : ganador(X)=b}).</code>

Las *reglas random con un solo atributo con cuerpo en la regla* tienen el siguiente formato:

`[Nombre_regla] random(Atributo) :- Cuerpo.`

donde `Nombre_regla` es un identificador para la regla, `Atributo` fue declarado en la sección de declaraciones, `Cuerpo` es un hecho que debe cumplirse para que la regla random sea verdadera, este tipo de regla puede no tener cuerpo.

Tabla 6. Se escogerá un valor aleatorio para el atributo `salida`.

<code>[r2] random(salida).</code>

Información probabilística. En P-log tiene el siguiente formato:

[Nombre_regla] pr(Función | Condición) = Valor.

donde Nombre_regla es un identificador para la regla y debe ser igual al Nombre_regla que asignamos en la regla random, Función es un valor que puede ser obtenido en las reglas random, Valor es un número entre [0, 1] escrito de forma Y/X, con X,Y dos números cualesquiera, y $X \geq Y$, Condición es un valor que debe ser cierto en las reglas random, esta condición puede ir o no.

Tabla 7. La probabilidad de que llegue en primer lugar el corredor 2 es de 1/5 (0.20).

[r1] pr(primero = 2) = 1/5.

Observaciones y acciones. Las observaciones y acciones son de la forma: obs(1). do(1). donde 1 es una literal. Las observaciones son usadas para llevar registro de los resultados de los eventos random. Por otro lado, las acciones indican que se hace verdadero como resultado de una acción deliberada, es decir, hubo una intervención sobre algún atributo Por ejemplo, si decimos que primero = 1 dentro del programa, entonces primero siempre será 1 y no podrá tener ningún otro valor y solo tendrá un mundo posible, en donde primero es igual a 1. Agregaremos una observación a este ejemplo:

Tabla 8. Agregamos la observación que la salida sea buena.

obs(salida=b).

Ya hemos mencionado que la finalidad de este ejemplo, es saber que probabilidad tiene cada uno de los corredores de llegar en primer lugar. Para esto se realiza una consulta al final del programa que sería de la forma siguiente:

Tabla 9. En esta tabla hacemos la consulta para saber que probabilidad hay de que el corredor 2 llegue en primer lugar.

? {primero = 2}.

Para terminar esta sección presentamos el programa completo escrito en P-log que modela el problema del corredor planteado al inicio de esta sección.

```
#domain corredor(C).
corredor= {1..4}.
#domain boolean(B).
boolean={b,f}.

salida : boolean.
primero : corredor.
ganador : corredor -> boolean.

ganador(C) = b :- not ganador(C) = f.
ganador(C) = f :- salida = b.

[r1] random(primero:{X: ganador(X)=b}).
[r2] random(salida).
```



```
[r1] pr(primero=2) = 1/5.
```

```
? {primero=2} | obs(salida=b).
```

Vemos la observación se agrega después de un signo “|” que va después de la consulta que se le hace al programa, en este caso preguntamos cual es la probabilidad que el ganador de la carrera sea el corredor 2 observando que la salida es buena.

3. Lenguaje P-log y su traducción a SMOBELS

En esta sección vamos a mostrar la traducción de un programa escrito en lenguaje P-log a un programa SMOBELS. Vemos importante realizar la traducción de un programa escrito en lenguaje P-log a un programa SMOBELS porque nos facilitará entender la semántica del lenguaje P-log.

Retomaremos el ejemplo del corredor visto en la subsección 2.2 y explicaremos cómo se realiza la traducción a SMOBELS, el orden de cómo explicar la traducción es conforme a las partes que estructuran un programa P-log.

Traducción de la parte de declaraciones: Como ya sabemos las declaraciones en un programa P-log puede tener una o todas de las siguientes formas: *definición de dominio de tipo*, *definición de tipos*, *declaración de tipos para atributos*, *declaración de la parte regular*.

Una *definición de dominio de tipo* en P-log tiene el siguiente formato: `#domain Tipo(Variable)`. y su traducción a SMOBELS tiene el mismo formato, es decir, este código es el mismo para P-log y SMOBELS. Por lo tanto, para nuestro ejemplo la traducción queda igual que en la tabla 1.

La *definición de tipos* en P-log puede ser de 2 formas:

- en forma de lista y tiene el siguiente formato: `Tipo = {X1, ... , Xn}`.
- en forma de rango y tiene el siguiente formato `Tipo = {X1 .. Xn}`.

Su traducción a SMOBELS sería lo siguiente:

- en forma de lista se traduciría a: `Tipo(X1). Tipo(X2). ... Tipo(Xn)`.
- en forma de rango se traduciría a: `Tipo(X1 .. Xn)`.

Para nuestro ejemplo, veremos la traducción en la siguiente tabla.

Tabla 10. En SMOBELS se escribe cada atributo por separado como un átomo con un solo argumento, resultaría más cansado si se tuvieran muchos numeros pues se tardaría mucho en escribirlo.

Lenguaje P-log	Traducción a SMOBELS
<code>corredor = {1 .. 4}.</code> <code>boolean = {b,f}.</code>	<code>corredor(1). corredor(2). corredor(3).</code> <code>corredor(4).</code> <code>boolean(b). boolean(f).</code>

La *declaración de tipos para atributos* en P-log no tienen una traducción a SMOBELS, sin embargo, son utilizados cuando se traducen las reglas random.

La *declaración de la parte regular* de un programa P-log consiste en una colección de reglas formadas usando literales. Éstas reglas pueden ir o no en el programa.

Para nuestro ejemplo, veremos la traducción en la siguiente tabla.

Tabla 11. En esta parte regular P-log la traducción se hace en átomos con 2 argumentos.

Lenguaje P-log	Traducción a SMOBELS
<pre>ganador(C) = b :- not ganador(C) = f. ganador(C) = f :- salida = f.</pre>	<pre>ganador(C,b) :- not ganador(C,f). ganador(C,f) :- salida(f).</pre>

Traducción de la parte de reglas de selección random. Como ya sabemos existen 2 formas de declarar una regla random en P-log: *declarando reglas random con un atributo y una función de condición, declarando reglas random un atributo y cuerpo en la regla.*

Las *reglas random con un solo atributo y una función de condición* tienen el siguiente formato:

```
[Nombre_regla] random(Atributo:{X:Función(X)}).
```

Su traducción a SMOBELS sería lo siguiente:

```
1{Atributo(X_): Rango(X_)}1 :- not intervene(Atributo).
pd( Nombre_regla, Atributo) :- not intervene(Atributo),
    Variable(X), Función(X).
:- not intervene(Atributo), Atributo(X), Función(X), Rango(X).
show Atributo(X_).
```

Para nuestro ejemplo, veremos la traducción en la siguiente tabla.

Tabla 12. Este tipo de reglas random es más complicado de traducir, pues una línea de código en P-log tiene 4 líneas de código en SMOBELS.

Lenguaje P-log	Traducción a SMOBELS
<pre>[r1] random (primero: {X: ganador(X) = b}).</pre>	<pre>1{primero(X) : corredor(X)}1 :- not intervene(primero). pd(r1,primero(X)) :- not intervene (primero), corredor(X), ganador(X,b). :-not intervene(primero), primero(X), not ganador(X,b), corredor(X). show primero(X).</pre>

Las reglas random no existen en SMOBELS, pero podemos intuirlo gracias a otras reglas que nos ayudan a realizar la misma función, la primera línea de código nos indica que primero solo tendrá un valor de tipo corredor siempre y cuando no haya intervención sobre primero. En la segunda línea de código ponemos pd para indicar que estamos hablando de probabilidad también no debe haber una intervención sobre primero, el valor que se asigne a primero debe ser de tipo corredor, además que la condición de ganador debe ser verdadera, show es una palabra reservada en SMOBELS que muestra el valor de primero.

Las reglas *random con un solo atributo con cuerpo en la regla* tienen el siguiente formato:

```
[Nombre_regla] random(Atributo) :- Cuerpo.
```

Su traducción a SMOBELS sería lo siguiente:

```
1{Atributo(X_): Rango(X_)}1:- not intervene(Atributo).
pd( Nombre_regla, Atributo) :- not intervene(Atributo),
                                Rango(X_), Cuerpo.
show Atributo(X_).
```

Para nuestro ejemplo, veremos la traducción en la siguiente tabla.

Tabla 13. En esta traducción la regla *random* no tiene cuerpo, y solo se crean 3 reglas más para SMOBELS, mencionaremos que si la regla tuviera cuerpo, se agregaría al final en el cuerpo de la regla que inicia con *pd*.

Lenguaje P-log	Traducción a SMOBELS
[r2] random(salida).	1{salida(X_) : boolean(X_)}1:- not intervene(salida). pd(r2,salida(X_)) :- not intervene(salida),boolean(X_). show salida(X_).

Traducción de la parte de información probabilística. En P-log tiene el siguiente formato:

```
[Nombre_regla] pr(Función | Condición) = Valor.
```

Su traducción a SMOBELS sería lo siguiente:

```
pa (Nombre_regla, Función, di_(Y,X)) :- Condición.
```

Tabla 14. Resulta sencilla la traducción porque es una línea, sólo es un átomo con 3 argumentos, primero el nombre de la regla, segundo la función y tercero la probabilidad.

Lenguaje P-log	Traducción a SMOBELS
[r1] pr(primeros = 2) = 1/5.	pa(r1, primeros(2), di_(1,5)).

Podemos deducir que se utiliza *pa* para denotar probabilidad y *di_* para denotar el valor de la probabilidad.

Traducción de la parte de observaciones y acciones. Su traducción a SMOBELS es la misma *obs(l). do(l).* donde *l* es una literal.

Para nuestro ejemplo, veremos la traducción en la siguiente tabla.

Tabla 15. La traducción de una observación genera una restricción en SMOBELS, y 4 reglas más que irán en todos los programas.

Lenguaje P-log	Traducción a SMOBELS
? {primeros=2} obs(salida=b).	:- not salida(b). hide. show pd(X,Y). show pa(X,Y,Z). show primeros(2).

La observación se traduce en una restricción en la cual se pide que la salida sea buena, hide es una palabra reservada que oculta valores de salida, y solo pide mostrar donde estén las palabras pd, pa y que muestre cuando el ganador sea el corredor 2.

A continuación mostramos el programa completo escrito P-log y su traducción a SMOBELS.

Lenguaje P-log	Traducción a SMOBELS
#domain corredor(C).	#domain corredor(C).
#domain boolean(B).	#domain boolean(B).
corredor = {1 .. 4}. boolean = {b,f}.	corredor(1). corredor(2). corredor(3). corredor(4). boolean(b). boolean(f).
primero : corredor. salida : boolean. ganador : corredor -> boolean.	
ganador(C) = b :- not ganador(C) = f. ganador(C) = f :- salida = f.	ganador(C,b) :- not ganador(C,f). ganador(C,f) :- salida(f).
[r1] random (primero: {X:ganador(X)=b}).	1{primero(X):corredor(X)}1 :- not intervine(primero). pd(r1,primero(X)):- not intervine(primero), corredor(X), ganador(X,b). :-not intervine(primero), primero(X), not ganador(X,b),corredor(X). show primero(X).
[r2] random (salida).	1{salida(X_) : boolean(X_)}1:- not intervine(salida). pd(r2,salida(X_)) :- not intervine(salida), boolean(X_). show salida(X_).
[r1] pr(primero=2) = 1/5.	pa(r1,primero(2),di_(1,5)).
? {primero=2} obs(salida=b).	:- not salida(b). hide. show pd(X,Y). show pa(X,Y,Z). show primero(2).

Los mundos posibles que se obtienen son:

Mundo posible 1: primero(4) salida(b)
Mundo posible 2: primero(2) salida(b)
Mundo posible 3: primero(3) salida(b)
Mundo posible 4: primero(1) salida(b)

La respuesta a la consulta de que probabilidad hay de que el corredor 2 sea el primero es de 0.20.

El programa de P-log se ejecutó en el sistema P-log diseñado por Weijun [4].

4. Conclusiones

Durante este artículo se explicó de forma muy detallada cómo se estructura P-log y cómo se traduce a SMOBELS una implementación de ASP.

Los programas hechos en el lenguaje P-log se pueden ejecutar en el sistema P-log⁵, un sistema creado por Weijun Zhu (2008) para obtener los mundos posibles.

Recordar que podemos encontrar más ejemplos de programas hechos en lenguaje P-log y su traducción a un programa SMOBELS, en la siguiente dirección web:

<https://sites.google.com/site/rmdaniel04/>

Podemos ver que la traducción se realiza de un programa P-log a un programa SMOBELS, pero un trabajo a futuro se podría realizar la traducción a otra implementación ASP como Potassco o DLV.

Agradecimiento. Este trabajo ha sido apoyado por el Proyecto de Ciencia Básica del Fondo Sectorial SEP-CONACyT con número de registro 101581.

Referencias

1. M. Gelfond, V. Lifschitz: The stable model semantics for logic programming. In R. Kowalski and K. Bowen, editores, 5th Conference on Logic Programming. MIT Press (1988)
2. Answer Set Programming, http://en.wikipedia.org/wiki/Answer_set_programming
3. David Pearce: Introducción a ASP (Answer Set Programming - programación con conjuntos respuestos) (2009)
4. C. Baral, M. Gelfond, N. Rushton: Probabilistic reasoning with answer set. Cambridge University Press (2008)
5. Tommi Syrjänen: Manual de usuario de Lparse, <http://www.tcs.hut.fi/Software/smodels/index.html#downloads>
6. Stable model semantics, http://en.wikipedia.org/wiki/Stable_model_semantics
M. Gelfond, V. Lifschitz: Action Languages. In: Electronic Transactions on Artificial Intelligence, vol. 2, pp 193-210 (1998)

⁵ <http://www.cs.ttu.edu/~wezhu/>

Algoritmos y aplicaciones

Implementación de una hiperheurística para la reducción de conflictos en el bloque quirúrgico

Marina I. Ramos-Martínez, Martín Carpio, Martín Soria-Alcaraz
y Héctor José Puga-Soberanes

División de Estudios de Posgrado e Investigación, Instituto Tecnológico de León,
León, Guanajuato, México
marramos85@gmail.com, jmcarpio61@hotmail.com,
soajorgea@gmail.com, pugahector@yahoo.com

Resumen A la fecha en los hospitales se requiere aprovechar los recursos del bloque quirúrgico, administrando de una forma más eficiente las salas para reducir el tiempo que el paciente tiene que esperar para ser atendido. Los procedimientos por lo general son organizados en papel y hojas de cálculo, lo cual contribuye a no hacer un uso eficiente del quirófano. Debido a esto, se convierte en una necesidad el desarrollo de una herramienta que permita distribuir las cirugías en el bloque quirúrgico de forma óptima. Con la implementación de un Árbol de Decisión para satisfacer las restricciones del bloque quirúrgico y una Hiperheurística la cual tiene como órgano rector un algoritmo memético para generar la combinación de tres heurísticas, se obtiene una solución que permite la optimización del bloque quirúrgico para un conjunto de instancias de prueba simuladas con distribución uniforme.

Palabras clave: Satisfacción de restricciones, árbol de decisión, algoritmo memético, hiperheurística.

1. Introducción

Gran parte de los problemas de logística hospitalaria son análogos a varios de los problemas que surgen en la industria y que son analizados bajo el enfoque de la optimización de recursos [1]. Por optimización se entiende al acto de obtener el mejor resultado posible dadas ciertas circunstancias [2]. La optimización tiene una enorme utilidad en muchas ramas del conocimiento. En la actualidad, no existe ningún método de optimización para resolver todo tipo de problemas, por lo que a lo largo de los años se han desarrollado diversos métodos para resolver distintas clases de problemas [3].

En los problemas donde el dominio (valores que definen al problema) es típicamente finito, puede parecer trivial encontrar una solución, ya que existe un proceso para determinar la solución óptima buscada, que es realizar una exploración exhaustiva del conjunto de soluciones. En teoría, este método puede llegar a la solución óptima, sin embargo puede llegar a tardar años o nunca encontrar la solución en tiempo razonable. Los algoritmos de tipo heurístico han sido empleados por su capacidad de proporcionar soluciones en tiempos razonables.

Con la implementación de algunas técnicas de Soft Computing, se creó una herramienta que permite obtener una Programación Quirúrgica inicial a partir de instancias de prueba simuladas con distribución uniforme. Las instancias de prueba manejan 3 prioridades para garantizar la selección de pacientes con mayor urgencia,

donde 1 es urgente, 2 moderado y 3 no urgente. La herramienta generada consta de una Hiperheurística que tiene como órgano rector un algoritmo memético y como base un Árbol de Decisión. Dicha herramienta genera la combinación de heurísticas de bajo nivel para maximizar el número de cirugías.

El uso de una herramienta para el control de pacientes en el bloque quirúrgico tiene un alto impacto en la eficiencia de las operaciones, principalmente en el ahorro de costos y la mejora del nivel de servicio ofrecido a los pacientes. El mayor desafío para el área quirúrgica es incrementar la eficiencia del hospital, de los recursos y la reducción en las listas de espera [4].

2. Marco Teórico

2.1. Satisfacción de Restricciones

La Satisfacción de Restricciones (CSP), es la operación de encontrar valores para las variables para satisfacer un conjunto de restricciones [5]. El modelo CSP permite representar de manera natural, y resolver de manera eficiente, numerosos problemas de Informática en general y de Inteligencia Artificial en particular [6]. La CSP trata con problemas de dominios finitos. La resolución de un CSP consta de dos fases diferentes [1], [7]:

- Modelar el problema como un problema de satisfacción de restricciones. La modelización expresa el problema mediante una sintaxis de CSPs, es decir, mediante un conjunto de variables, dominios y restricciones del CSP.
- Procesar el problema de satisfacción de restricciones resultante. Una vez formulado el problema como un CSP, hay dos maneras de procesar las restricciones:
 - 1) Técnicas de consistencia. Se basan en la eliminación de valores inconsistentes de los dominios de las variables.
 - 2) Algoritmos de búsqueda. Se basan en la exploración sistemática del espacio de soluciones hasta encontrar una solución o probar que no existe tal solución.

Un problema CSP es una terna (X, D, C) donde X es un conjunto de n variables $\{X_1, \dots, X_n\}$.

$D = \langle D_1, \dots, D_n \rangle$ es un vector de dominios. La i -ésima componente D_i es el dominio que contiene todos los posibles valores que se le pueden asignar a la variable X_i .

C es un conjunto finito de restricciones. Cada restricción n -aria (C_n) está definida sobre un conjunto de variables $\{X_1, \dots, X_n\}$ restringiendo los valores que las variables pueden simultáneamente tomar.

Una solución a un CSP es una asignación de valores a todas las variables de forma que se satisfagan todas las restricciones [8]. Es decir, una solución es una tupla consistente que contiene todas las variables del problema. Por tanto un problema es consistente, si existe al menos una solución, es decir una tupla consistente (a_1, a_2, \dots, a_n) .

Los objetivos que deseamos obtener de un CSP se centran en encontrar:

- Una solución, sin preferencia alguna.
- Todas las soluciones.

- Una óptima, o al menos una buena solución dando alguna función objetivo definida en términos de algunas o todas las variables.

Existen dos tipos de restricciones [6]:

- Restricciones Duras. Son condiciones de obligatorio cumplimiento.
- Restricciones Blandas. Son restricciones que en realidad denotan preferencias del usuario y se desea que se cumplan en la medida de lo posible.

En esta investigación se implementó un algoritmo de búsqueda como lo es el Árbol de Decisión para la satisfacción de restricciones.

2.2. Árbol de decisión

Los Árboles de Decisión (AD) han sido usados desde hace muchos años en diversas disciplinas, como estadística, ingeniería, la teoría de decisión, etc. En todas esas áreas de aplicación, los árboles de decisión han sido utilizados para hacer exploración de datos para cubrir alguno de los siguientes objetivos: para representar el conjunto de observaciones de una manera más concisas y eficiente, para descubrir si el conjunto de ejemplos contiene agrupaciones de objetos claramente separables, para descubrir una función entre las variables independientes o atributos, entre otros [1], [9].

Un AD construye diagramas basados en reglas, que sirven para representar y categorizar una serie de condiciones que ocurren de forma sucesiva, para dar solución a un problema dado. Los valores que pueden entrar o salir del árbol pueden ser discretos o continuos. Los árboles de decisión permiten extraer y representar reglas de clasificación que subyacen a un conjunto de ejemplos observados mediante una estructura jerárquica secuencial que particiona recursivamente el conjunto de ejemplos. En un AD cada camino desde la raíz hasta las hojas representa una regla conjuntiva de clasificación. Cada nodo contiene una pregunta sobre un determinado atributo, el arco hacia el nodo siguiente un valor posible para este atributo y la hoja final una predicción sobre la clase que debe asignarse a los ejemplos que cumplan las condiciones establecidas en el camino [1], [9].

Los AD permiten satisfacer las restricciones [7], en este caso las restricciones en el problema de la Programación Quirúrgica, sin embargo necesita de métodos metaheurísticos para lograr la optimización del bloque quirúrgico.

2.3. Algoritmos meméticos

Los algoritmos meméticos (MA), son metaheurísticas basadas en población. Esto quiere decir que mantienen un conjunto de soluciones candidatas para el problema considerado [10]. Los orígenes de los MA se remontan a finales de los años ochenta y deben su nombre al concepto de “meme” introducido por R. Dawkins [11]. La idea básica de los MA es combinar conceptos y estrategias de diferentes metaheurísticas para aunar las ventajas de las mismas. El principio de operación de esta metaheurística es el siguiente: “De la misma forma que en una población se transmiten los genes de los padres a los hijos, los memes se transmiten de cerebro a cerebro de la población” [11]. Desde un punto de vista algorítmico, es común considerar al MA como Algoritmos Genéticos al que se añade un procedimiento de búsqueda local.

Los procedimientos de búsqueda local son sencillos métodos de mejora, en los que en cada iteración se sustituye la solución por una solución vecina mejor [12]. Más concretamente en este caso, sea S una solución, definimos el vecindario de S , $N(S)$, como sigue:

Es decir, el conjunto de soluciones que se obtiene intercambiando un elemento que está fuera de S con otro que está dentro. Nuestro procedimiento de búsqueda local se describe en el algoritmo 2.1:

Algoritmo 2.1 Algoritmo memético

```

1 Leer solución inicial  $S$ 
2 Repetir
3 Hacer  $valor\_ant=f(S)$ 
4   Buscar  $f(S^*)= \max \{f(S')/ S' \in N(S)\}$ 
5   Si  $f(S^*) > f(S)$  entonces hacer  $S=S^*$ 
6   Hasta  $f(S^*) \leq valor\_ant$ 

```

La idea central de los MA subyace a mejoras individuales de las soluciones en cada uno de los agentes junto con procesos de cooperación y competiciones de tipo poblacional [13].

2.4. Hiperheurística

El proceso de investigación muestra que es posible la implementación de Hiperheurísticas (HH) para tratar de obtener la optimización del bloque quirúrgico. Las HH han surgido como parte de un nuevo enfoque. El término HH fue introducido por Cowling, en donde las define de la manera siguiente:

Enfoques que funcionan a un nivel más alto de abstracción que las heurísticas y gestionan la elección de lo que debería ser un método heurístico de bajo nivel para ser aplicado en un momento dado, dependiendo de las características de la región del espacio de soluciones en exploración [14].

A partir de esta definición, un algoritmo hiperheurístico básico se muestra en 2.2:

Algoritmo 2.2 Hiperheurística

```

/* Conjunto de heurísticas */
1  $H = \{h_1, h_2, \dots, h_m\}$ 
2  $i \leftarrow 0$ 
3 while (Criterio de terminación no satisfecho) do
4   Seleccionar la heurística  $h_j$  más adecuada, donde  $j \in \{1, 2, \dots, |h|\}$ 
5   Transformar el estado actual Si al estado  $S_{i+1}$  aplicando  $h_j$ 
6  $i \leftarrow i + 1$ 

```

La idea básica que reside detrás de las HH consiste en utilizar un conjunto de métodos heurísticos que actúan sobre un problema determinado (heurísticas de bajo

nivel), de tal forma que en cada momento la HH decida cuál es el mejor de ellos para resolver el problema. En otras palabras, las HH operan en el espacio de las heurísticas mientras que las heurísticas trabajan directamente sobre el espacio de las soluciones del problema [15].

La HH no es la encargada de buscar la mejor solución al problema. Lo que hace, es seleccionar en cada paso del proceso de búsqueda las heurísticas de bajo nivel más prometedoras que sean potencialmente capaces de mejorar la solución actual. Por otra parte, si no hay mejora, la HH debe ser capaz de diversificar la búsqueda a otras regiones mediante la selección de otras heurísticas, buscando de esta manera poder escapar de óptimos locales.

Las HH trabajan a un nivel más alto de abstracción y comúnmente no incluyen información alguna sobre el dominio del problema, por lo que son desarrollados como métodos de optimización general y de esta manera pueden aplicarse a una gama más amplia de problemas. Idealmente, una HH sólo requiere como parámetros el conocer el número de heurísticas de bajo nivel así como el tipo de función objetivo a optimizar (mín $f(x)$ o máx $f(x)$) [15].

Las HH pueden ser descritas como estrategias diseñadas para controlar la aplicación de un conjunto de heurísticas durante el proceso de búsqueda y suelen verse como cajas negras (Figura 1) [16].

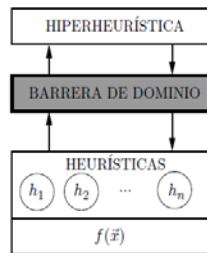


Figura 1. Marco de una HH tomado de [15].

Actualmente existen diversos mecanismos de selección diseñados para las HH, los cuales, según Chakhlevitch y Cowling [15], se pueden clasificar en cuatro grandes grupos:

1. Hiperheurísticas basadas en selección aleatoria
2. Hiperheurísticas codiciosas
3. Hiperheurísticas basadas en metaheurísticas
4. Hiperheurísticas con aprendizaje

Se proponen 3 características del enfoque hiperheurístico [17][18] que aparecen implícitas en la mayoría de las aplicaciones, mas no son definidas explícitamente, éstas características se enuncian a continuación:

a) Repetitividad de la implementación de la solución otorgada: el resultado de una HH es un algoritmo que soluciona n instancias de un problema dado, es necesario entonces que la solución HH posea una estructura que sea fácilmente aplicable a diversas instancias y resolver con un rendimiento adecuado.

b) Dirección en la búsqueda de solución: el espacio de búsqueda del enfoque hiperheurístico al manejar una representación no directa podrá ser menor que el espacio de búsqueda original del problema, aunque el espacio de búsqueda nunca es en realidad disminuido, el proceso de evaluación si se acelera al dirigir la búsqueda.

c) Adquisición de experiencia por parte del investigador: la respuesta es un algoritmo de solución, este algoritmo definido como el conjunto de heurísticas que resuelven un problema, puede ser analizado por el investigador, denotando pues elementos claves que resuelven su problema, aumentando así su experiencia personal, a diferencia de un enfoque metaheurístico donde se resuelve el problema sin la información del cómo se resolvió.

3. Implementación del árbol de decisión y la hiperheurística

Se implementó una heurística para la organización de las cirugías, la cual consiste en agendar la cirugía programada en el día y turno del médico especialista correspondiente, asignado la primera cirugía independientemente del tiempo que dure en la sala 1, la segunda cirugía en la sala 2 y posteriormente la tercera cirugía en la sala 3. La finalidad de la heurística es distribuir las cirugías en las 3 salas de tal forma que no queden salas vacías. Como restricción de programación se van sumando los tiempos por sala, día y turno, si al sumar la nueva cirugía excede en más de 360 minutos está no puede ser agendada en esa sala por lo que busca si puede ser programada en la siguiente sala y así sucesivamente. En caso de no encontrar una sala disponible ese día busca en el siguiente día que opere el médico especialista, en caso de no poder ser asignada en otro día, se crea una nueva semana y se sigue el mismo procedimiento. El siguiente algoritmo 3.1 [1] muestra como se programan las cirugías:

Esta heurística permite la satisfacción de las restricciones. Para la optimización del bloque quirúrgico se implementó una HH.

La parte medular del enfoque hiperheurístico es el conjunto de heurísticas usadas a fin de crear un algoritmo de solución, se eligieron 3 heurísticas para ser implementadas en la HH, dichas heurísticas son [1]:

- Heurística 1 Orden Ascendente: Esta heurística consiste en ordenar la instancia de prueba de acuerdo al tiempo de las cirugías, seleccionar las cirugías una por una de menor a mayor tiempo para ser programada de ser posible en alguna de las salas.
- Heurística 2 Orden Descendente: Esta heurística consiste en ordenar la instancia de prueba de acuerdo al tiempo de las cirugías, seleccionar las cirugías una por una de mayor a menor para ser programada de ser posible en alguna de las salas.
- Heurística 3 Ordenar Mediante la Mediana: Esta heurística consiste en ordenar la instancia de prueba de acuerdo al tiempo de las cirugías, seleccionar las cirugías una por una de acuerdo a la mediana para ser programada de ser posible en alguna de las salas.
- La HH consiste en generar una mezcla de las 3 heurísticas anteriores, de tal forma que en cada iteración se busque la combinación de estas que genere mejores resultados, es decir, que dé como resultado un mayor número de cirugías programadas por semana. La HH consta de 2 fases, una de entrenamiento y una de aplicación. En el entrenamiento se obtiene la combinación de heurísticas de bajo nivel que de como resultado un mayor número de cirugías, mientras que en la fase de aplicación se genera la Programación Quirúrgica a partir de la solución

Algoritmo 3.1 AD para la programación quirúrgica

```
1 Mientras existan nuevos registros
2 Inicializar población
   Día de operación del médico especialista
   Tiempo promedio de la cirugía
3 Si médico especialista opera día (lunes, martes, miércoles, jueves, viernes, sábado o
   domingo)
4 Tiempo de nueva cirugía + tiempo acumulado sala 1 < 360
5 Nueva cirugía <-- sala 1
6 Si no
7 Tiempo de nueva cirugía + tiempo acumulado sala 2 < 360
8 Nueva cirugía <-- sala 2
9 Si no
10 Tiempo de nueva cirugía + tiempo acumulado sala 3 < 360
11 Nueva cirugía <-- sala 3
12 Si no
13 Buscar otro día que opere el médico especialista regresar al paso 2
14 Si no
15 Crear nueva semana regresar al paso 2
16 Fin mientras
```

encontrada en el entrenamiento, esto independientemente que las instancias de prueba no sean las mismas que con las que se entreno (barrera de dominio).

- La HH implementada se basa en el MA, donde primero se genera la población inicial para evaluar la función. La población es inicializada con n valores aleatorios entre 1 y 3 (genes), donde cada gen representa una heurística de bajo nivel. La variable n representa el tamaño de la instancia de prueba.
- La población es evaluada para calcular el fitness (cantidad de cirugías programadas), los fitness son ordenados y el mayor resultado representa la primera solución. Para formar la nueva generación, el padre y la madre son seleccionados mediante una ruleta para realizar la cruce y formar a los dos descendientes. Los dos hijos son mutados con cierta probabilidad, es decir, se cambia un gen al azar del individuo. Posteriormente se evalúan, el que tenga el mayor fitness es seleccionado para formar parte de la nueva generación (figura 2).

La figura 2 muestra el funcionamiento interno de la HH.

4. Pruebas y resultados

Para evaluar la solución generada por la HH (es decir la combinación de heurísticas de bajo nivel que permitiera obtener un mayor número de cirugías programadas), se

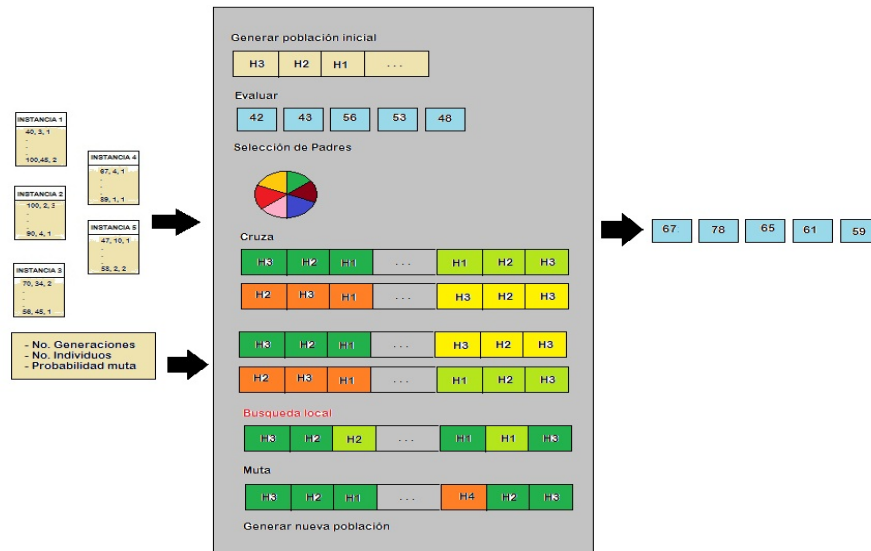


Figure 2. Diagrama conceptual de la HH con MA como órgano rector.

generaron 10 instancias de prueba con distribución uniforme, las cuales cuentan con 150 registros cada una. Cada registro tiene 3 valores, el primer valor es el tiempo estimado de la cirugía que va desde 60 a 360 minutos, el segundo la clave del médico especialista y por ultimo el grado de prioridad 1, 2 ó 3. Se realizaron un total de 10 pruebas, donde en cada prueba se generó una solución a partir de seleccionar 5 instancias aleatorias y correr el MA en 20,000 generaciones (Entrenamiento). Posteriormente se aplicó la solución generada por la HH en el entrenamiento a las 5 instancias de prueba restantes seleccionadas en forma aleatoria uniforme (Aplicación). Como se puede observar en la tabla 1, la cantidad de cirugías programadas es muy cercana a la cantidad máxima encontrada por la mejor heurística de bajo nivel. En la figura 3 se muestra una gráfica comparativa de los resultados obtenidos para las 10 instancias en cada una de las pruebas, donde la serie 11 es el mejor valor encontrado por las heurísticas de bajo nivel.

5. Conclusiones

La optimización de la Programación Quirúrgica es un problema que a la fecha tienen algunos hospitales. Para el caso de estudio analizado, aun es un gran problema, ya que actualmente realizan la programación manualmente con la ayuda de herramientas de cálculo. Esto genera inconsistencia en los datos, pérdida de información e incluso un mal aprovechamiento del bloque quirúrgico.

La HH logró generar la combinación de heurísticas de bajo nivel que permitió la selección del mayor número de cirugías tomando en cuenta toda la lista destinada a programar. La herramienta diseñada permite encontrar una solución en tiempo real, sin embargo por ser un método heurístico no garantiza encontrar la solución óptima.

Tabla 1. Pruebas simuladas

Prub	Valor Encontrado por las Heurísticas										Valor Encontrado por la HH										Entrena	Aplica	Dif 1	Dif 2
	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10				
1	71	77	68	64	74	70	71	80	68	69	70	73	67	66	70	69	67	79	63	64	4 6 2 1 8	5 1 0 7 9 3	5	-1
2	71	77	68	64	74	70	71	80	68	69	68	75	65	65	73	67	67	76	64	65	2 1 9 5 4	1 0 7 3 8 6	9	2
3	71	77	68	64	74	70	71	80	68	69	65	74	65	63	64	65	69	79	65	66	1 0 2 9 8 7	3 4 5 6 1	12	8
4	71	77	68	64	74	70	71	80	68	69	68	75	67	65	72	66	66	77	65	61	4 9 5 2 3	1 0 1 7 6 8	7	7
5	71	77	68	64	74	70	71	80	68	69	68	74	66	65	69	66	71	78	64	65	1 7 4 9 2	3 6 1 0 8 5	9	-1
6	71	77	68	64	74	70	71	80	68	69	65	75	64	65	65	67	70	79	64	65	2 4 9 7 8	1 3 1 0 5 6	7	7
7	71	77	68	64	74	70	71	80	68	69	64	75	65	65	66	69	69	79	64	62	7 2 8 4 6	1 0 1 5 3 9	5	8
8	71	77	68	64	74	70	71	80	68	69	70	76	68	64	72	66	67	78	63	67	2 5 3 1 0 1	8 6 7 9 4	6	1
9	71	77	68	64	74	70	71	80	68	69	69	76	66	62	72	66	65	79	65	66	9 1 0 8 5 2	1 7 6 4 3	10	1
10	71	77	68	64	74	70	71	80	68	69	69	76	66	64	70	66	67	77	66	66	5 9 8 1 0 2	7 3 6 4 1	13	-1

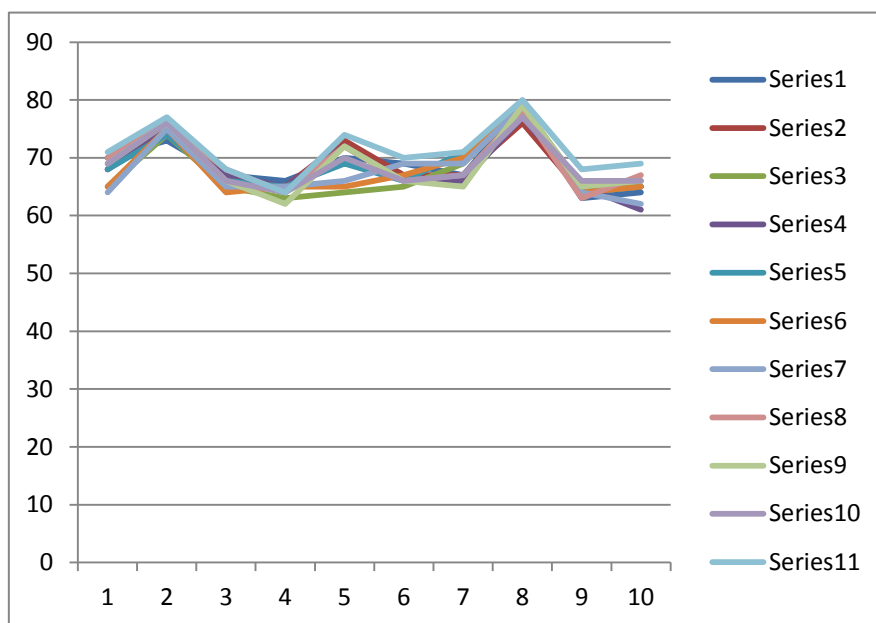


Figura 3. Distribución de las cirugías.

Como trabajo futuro se propone el uso de esta herramienta para generar una agenda dinámica que permita tener un mayor control sobre los procedimientos quirúrgicos, dando como resultado un mayor aprovechamiento del bloque quirúrgico. Dicha herramienta puede ser ligada a un servidor para poder ser monitoreada desde cualquier dispositivo móvil ó sistema de cómputo para que el especialista esté informado en

tiempo real de lo que ocurre en el bloque quirúrgico y sepa la hora en que le toca operar al siguiente paciente.

Agradecimientos. Al Instituto Tecnológico de León y a CONACyT por el apoyo brindado para el desarrollo de la investigación tecnológica y científica.

Referencias

1. Marina I Ramos: Implementación de Heurísticas Para la Reducción de Conflictos en el Bloque Quirúrgico. Instituto Tecnológico de León. IX Encuentro Participación de la Mujer en la Ciencia, León Guanajuato (2012)
2. Rao, S. S.: Engineering Optimization, Theory and Practice. Third ed. John Wiley & Sons, Inc. (1996)
3. Köppen M.: No-Free-Lunch Theorems and the Diversity of Algorithms. Pascalstr 8-9, 10587 Berlin, Germany (2004)
4. C. Núñez: La información del bloque quirúrgico, clave de sanidad para el siglo XXI. TIC en sanidad, No. 248, pág. 432-433 (2008)
5. Alfonso, María: Un modelo de integración de técnicas de Clausura y CSP de restricciones temporales: Aplicación a problemas de Scheduling. Tesis Médico especialista, Universidad de Alicante (2001)
6. Larrosa Javier: Restricciones Blandas: Modelos y Algoritmos. Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial. No. 20 (2003)
7. Barber, Federico: Introducción a la Programación de Restricciones. Departamento de Sistemas Informáticos y Computación. Universidad Politécnica de Valencia, España (2004)
8. Mánya Felip: Técnicas de Resolución de Problemas de Satisfacción de Restricciones. Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial, vol. 7 número 019. España (2003)
9. Joaquim Llisteri Boix: Tratamiento del lenguaje natural: tecnología de la lengua oral y escrita. Fundación Duques de Soria (2002)
10. Cotta, Carlos: Una Visión General de los MA. Dept. Lenguajes y Ciencias de la Computación. ETSI Informática, University of Málaga, España (2004)
11. L. Meléndez, Valadés: Metaheurísticas. Editorial DYKINSON, Universiad Rey, Juan Carlos, Madrid (2007)
12. Bonrostro, Joaquín: Algoritmo Memético para Selección de Variables en el Análisis Discriminante. Departamento de Economía Aplicada. Universidad de Burgos (2004)
13. Moscato, Pablo: Una Introducción a los MA. School of electrical Engineering and Computer Science. University of Newcastle Callaghan, Australia (2002)
14. Cowling, P., Kendall, G., and Soubeiga, E.: A hyperheuristic approach to scheduling a sales summit. In: Proceedings of the Third Metaheuristic International Conference (MIC 2001), Porto, Portugal, vol. 2079, Springer-Verlag, pp. 127–131 (2000)
15. Chakhlevitch, K., and Cowling, P.: Hyperheuristics: Recent Developments. Vol. 136 of Studies in Computational Intelligence. Springer Berlin, pp. 3–29 (2008)
16. Villela Tinoco, Carlos: Mecanismo de selección y control de una hiperheurística basada en Evolución Diferencial para optimización en espacios restringidos.
17. Soria Alcaraz, Jorge Alberto: Diseño de horarios con respecto al alumno mediante técnicas de cómputo evolutivo. Instituto Tecnológico de León (2010)
18. Soria-Alcaraz Jorge A, Carpio Martin, Terashima-Marin Hugo.: Academic Timetabling Design Using Hyper-Heuristics. Soft Computing for Intelligent Control and Mobile Robotics. Studies in Computational Intelligence, Springer Berlin /Heidelberg, Vol 318, 43-56 (2012)

Modelado con selección de estructura automática para sistemas no lineales usando SVM difusos

Julio César Tovar Rodríguez

ESIME Zacatenco, Instituto Politécnico Nacional (IPN),
Av. IPN s/n, Lindavista, Gustavo A. Madero, C.P. 07738, México DF, México
jctovar@ipn.mx

Resumen En este trabajo se estudia la identificación de sistemas dinámicos no lineales usando redes neuronales difusas. Se enfoca en ambas incertidumbres estructurales y paramétrica las cuales han sido exploradas considerablemente en la literatura de identificación de sistemas no lineales. La contribución principal es que la estructura analítica integral se propuso para la selección de estructura de redes neuronales difusas, los parámetros de identificación y el intercambio de histéresis en la red garantiza una ejecución de identificación neuronal. Primeramente, un support vector machine automática, se propone dentro de un intervalo de tiempo fijo para un criterio dado de construcción de la red. Luego la actualización de los parámetros de la red garantiza un error acotado. La incertidumbre se hace frente con una estrategia de histéresis para habilitar el intercambio del identificador neuro difuso que garantiza la ejecución de la red a lo largo del proceso de intercambio. El análisis teórico y el ejemplo de simulación muestran la eficacia del método propuesto.

Palabras clave: Red neuronal artificial, *Support Vector Machine* (SVM), lógica difusa, identificación, histéresis.

1. Introducción

Las redes neuro difusas han sido usadas ampliamente en predicción de series de tiempo, modelado de sistemas no lineales y control [1]. Este trabajo se enfoca en el modelado de sistemas no lineales dinámicos con múltiples redes neuro difusas. El problema principal del modelado difuso es la extracción de las reglas, estas se pueden dividir dentro de dos clases [2]: 1) la obtención de las reglas de expertos [3], 2) la obtención de las reglas difusas automáticamente de los datos observados. El método experto usa el criterio imparcial y la técnica de prueba y error, está se aplica fuera de línea.

El proceso de la extracción de reglas difusas para el modelado de sistemas no lineales es llamado estructura de identificación. Un método común es la partición de los datos de entrada y salida, a esto se le conoce como cuadrícula difusa. Estos enfoques requieren que los datos estén listos antes del modelado. Existen pocos métodos de agrupamiento en línea en la literatura.

La idea básica del modelado SVM es el mapeo de las entradas dentro de un espacio característico de dimensión superior, entonces se resuelve la programación cuadrática (QP) con una función de costo apropiada. Solo las soluciones no zeros los cuales son

llamados support vectors son útiles para el modelado. A la combinación de multimodelos y redes neuronales difusas se les considera como un enfoque racional para identificación de sistemas no lineales. Se concluye que en sistemas en lazo cerrado son globalmente estables en el sentido que todas las señales involucradas están uniformemente acotadas [4].

Se verá una elección de estructura automatizada se propone primero dentro de un intervalo de tiempo fijo para un criterio de construcción de la red dada. Entonces el algoritmo de actualización de los parámetros de la red se propone con el error de identificación acotado garantizado. Para cubrir con la estructura de incertidumbre, una estrategia de histéresis se desarrolla para habilitar la conmutación del identificador neuro difuso con la ejecución garantizada a lo largo de la conmutación del proceso.

2. Múltiple redes neuronales difusas para identificación de sistemas no lineales

Una planta no lineal discreta puede ser representada por

$$y(k) = f[x(k), \theta] + e(k) \quad (1)$$

donde

$$\begin{aligned} x(k) &= [y(k-1), y(k-2), \dots, y(k-n_y), \\ &u(k-d), u(k-d-1), \dots, u(k-d-n_u)]^T \in \mathbb{R}^{m \times 1} \end{aligned} \quad (2)$$

y $f(\cdot)$ es una ecuación diferencial no lineal desconocida representando las dinámicas de la planta, $u(k)$ y $y(k)$ son escalares medibles entrada y salida, d es el retardo, θ es un vector de parámetros desconocidos asociado con una estructura del modelo apropiada, $e(k)$ es un ruido de observación acotado, n_y y n_u son las longitudes de salida y entrada, $n_y + n_u = n$. De hecho, (1) es un modelo NARX [2].

Un modelo difuso genérico se presenta como una colección de reglas difusas en la siguiente forma

$$\begin{aligned} R^j : & \text{IF } x_1 \text{ is } A_1^j \text{ and } x_2 \text{ is } A_2^j \text{ and } \dots x_n \\ & \text{is } A_n^j \text{ THEN } y \text{ is } B^j \end{aligned} \quad (3)$$

Se usa $l(j=1, 2, \dots, l)$ reglas difusas IF-THEN para realizar un mapeo del vector lingüístico de la entrada $x=[x_1, \dots, x_n] \in \mathcal{R}^n$ a la salida \hat{y} . Se sabe, usando el producto inferencia, promedio de centros y el difusificador singleton, la p -ésima salida del sistema lógico difuso se puede expresar como

$$y = \left(\sum_{j=1}^l w_j \left[\prod_{i=1}^n \mu_{A_i^j} \right] \right) / \left(\sum_{j=1}^l \left[\prod_{i=1}^n \mu_{A_i^j} \right] \right) = W \phi[V, x(k)] \quad (4)$$

Generalmente las redes neuronales difusas (4) no puede coincidir dado un sistema no lineal (1) exactamente, el sistema no lineal (1) se puede representar como:

$$y(k) = W\phi[V, x(k)] + \varepsilon(k) \quad (5)$$

donde $\varepsilon(k)$ se define como el error de modelado. El sistema no lineal (1) puede ser escrita como

$$y(k) = W^0\phi[V^0, x(k)] + f(k) \quad (6)$$

donde $\varepsilon(k)$ es el error modelado, V^0 y W^0 son conjuntos de $f(k)$ parámetros conocidos escogidos por el usuario.

De (6) se conoce el error modelado f_i depende de la estructura de la red neuronal difusa. Para algunos procesos no lineales, sus condiciones de operación varían con el tiempo o su entorno de operación es complicado, y un modelo no es suficiente para describir la planta completa. Los múltiples modelos pueden dar una mejor exactitud de identificación. Aunque una simple red neuronal difusa (4) puede identificar cualquier proceso no lineal (caja negra), el error de identificación puede ser grande si la estructura de la red difusa no se escoge apropiadamente. Generalmente hablando, no se puede encontrar la estructura de la red óptima representando el sistema (1) bajo todas las condiciones diferentes de operación. Una solución posible es usar varias redes y seleccionar la mejor por el algoritmo de conmutación apropiada. La estructura de los identificadores múltiples neuronales se muestra en la figura 1.

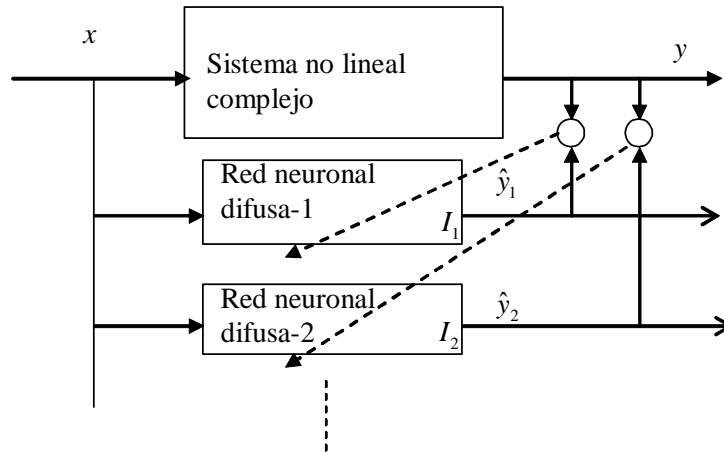


Fig. 1. Identificador múltiple neuro difuso

Aquí I_1, I_2 son los dos identificadores neuro difuso, cuyas salidas son \hat{y}_1, \hat{y}_2 . Se usa una política de conmutación para escoger un identificador I_i tal que el error de identificación entre la salida de esta red neuronal y la planta ($\hat{y}_i(k) - y(k)$) se minimiza. Los identificadores múltiples neuro difusos son presentados como

$$I_\sigma : y_\sigma(k) = W^\sigma(k)\phi[V^\sigma(k)x(k)] \quad (7)$$

donde $\sigma = \{1, 2, \dots, r\}$ y r son el número total de identificadores neuro difusos.

El objetivo de la identificación múltiples redes neuro difusas es diseñar redes neuro difusas apropiadas y la política de conmutación tal que un índice de ejecución es minimizado y la conmutación infinita no ocurre.

3. Selección de estructura automática

Un SVM puede separar los datos dentro de dos clases con un hiperplano de margen máximo [4]. Si el entrenamiento es separable por el hiperplano, la función es escogida como $f(x)=(w \cdot x)+b$. El margen se define como la distancia mínima desde una muestra hasta la superficie de resolución. El margen en turno se puede medir por la longitud del vector w , tal que los puntos cercanos al hiperplano satisface $|(w \cdot x)+b|=1$.

Para N muestras $\{x(k),y(k)\}_{k=1}^N$, se usa SVM para aproximar una función no lineal. Considerar la regresión en un conjunto de funciones no lineales

$$f(x) = w^T \phi(x) + b \quad (8)$$

donde el kernel $K(x,x_k)=\phi(x)^T \phi(x_k)$. Existen muchas posibilidades para escoger el kernel $K(x,x_k)$, el cual requiere la condición de Mercer [4]. Por ejemplo kernel lineal $K(x,x_k)=x^T x_k$, kernel MLP $K(x,x_k)=\tanh(k_1 x^T x_k + k_2)$, kernel RBF $K(x,x_k)=\exp(-\|x-x_k\|^2/\sigma^2)$. En esta sección, se usará el kernel difuso el cual se define como $K(x,x_k) = \{u(x_i, x_j)\}$, $i, j = 1, \dots, N$

donde u_i es una función de membresía. Se usa la función Gaussiana:

$$u(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \quad (9)$$

La regresión es resolver el problema siguiente

$$\begin{aligned} \min J_p &= \frac{1}{2} w^T w + C \sum_{k=1}^n s_k \xi_k \\ \text{sujeto a } &|y_k - (w^T \phi(x_k) + b)| \leq \varepsilon \end{aligned} \quad (10)$$

donde $\xi_k \geq 0$, $k=1,2,\dots,N$, C es constante, ξ_i es la variable débil. s_k es un factor difuso. Este denota el grado de importancia de la muestras x_i para el aprendizaje del hiperplano óptimo en SVM. Se selecciona si como la función forma de campana ver Figura 2. Así el s_k más pequeño es, el más pequeño efecto que la muestra x_k .

4. Selección de modelos

Se define el índice del error de desempeño $J_p(k)$ para el p-ésimo identificador neuronal como

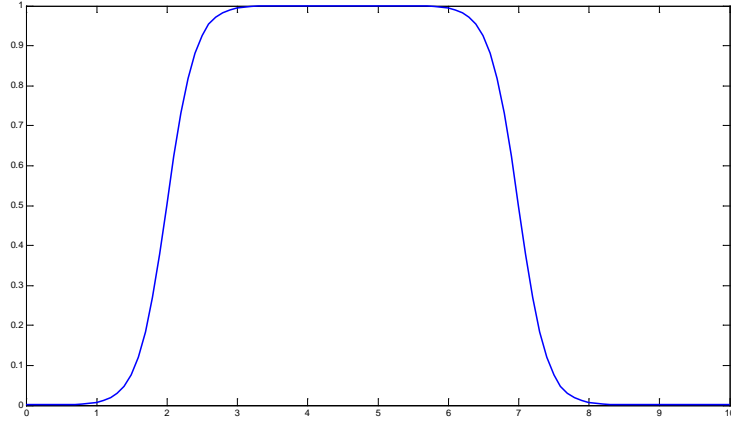


Fig. 2. Función de membresía en forma de campana generalizada.

$$J_p(k) \square \alpha e^2(k) + (1 - \alpha) \sum_{t=k-N_p}^k e^2(t) \quad (11)$$

donde α es un parámetro de diseño para ambos errores de pesos término corto y término largo, N_p es la longitud del p-ésimo identificador neuro difuso. La decisión de la estructura de conmutación (cambio) está dirigido por el monitoreo del índice de desempeño (11). Para prevenir un cambio rápido arbitrariamente debido a las perturbaciones, un algoritmo de decisión de histéresis es necesario. En este trabajo, se cambiara el modelo de la red neuro difusa solo cuando las funciones de membresía son casi convergentes. Se define la función de cambio como ($k > N$)

$$\omega(k) \square \frac{1}{2} \left| \text{tr}(W^T(k)W(k)) - \frac{1}{k-N} \sum_{t=N}^k \text{tr}(W^T(t)W(t)) \right| \quad (12)$$

$$\frac{1}{2} \left| \text{tr}(V^T(k)V(k)) - \frac{1}{k-N} \sum_{t=N}^k \text{tr}(V^T(t)V(t)) \right|$$

donde $V^T(k) = [c_{ji}(k), \sigma_{ji}(k)]$. El algoritmo de decisión de histéresis puede ser formulado como

$$\begin{cases} r+1 & \text{si } J(k) > J(k-1) + h \text{ y } \omega(k) \leq L \\ r & \text{otra manera} \end{cases} \quad (13)$$

donde $h > 0$ es la constante de histéresis, L es el umbral para el cambio de pesos. $r = r+1$ define la estructura de la r-ésima red neuronal no es apropiada para el dato entrante, se debe comenzar una red neuronal, ahora $\sigma = \{1, 2, \dots, r, r+1\}$. Se escoge L tal

que el algoritmo de conmutación (13) pueda trabajar después de los parámetros no afecten significativamente la identificación neuro difusa.

En general incluso para sistemas lineales conmutados invariantes en el tiempo, la estabilidad de cada componente del sistema en lazo conmutado no garantiza la estabilidad del sistema conmutado entero bajo leyes de conmutación arbitrarios. Algunas condiciones sobre la política de conmutación es muy especial: solo pueden conmutar desde el modelo r hasta el modelo $r+1$ en el eje de tiempo. La conmutación es una secuencia. El teorema asegura que el error de identificación en cada subsistema es estable. Puesto que los subsistemas son conmutados uno por uno, se discutirá la convergencia de nuestro esquema de identificación multi neuronal.

Lema 1. (a) Para cada tiempo finito k_1 , existe por lo menos un modelo $r_0 \in \sigma$ tal que el índice de desempeño $J_{r_0}(k)$ en (11) está acotado en $[1, k_1]$

(b) Para cada $i \in \sigma$ el índice de desempeño $J_i(k)$ en (11) tiene un límite (puede ser infinito) como $k \rightarrow k_1$. Sea f una función está definida sobre un intervalo abierto conteniendo a c (excepto posiblemente a c .) EL argumento "Límite Infinito" $\lim_{x \rightarrow c} f(x) = \infty$ significa que para cada $M > 0$ existe una δ tal que $f(x) > M$ siempre que $0 < |x - c| < \delta$.

Teorema 1. Si se usan las multi redes neuronales como en (7) y el algoritmo de conmutación de histéresis como en (13), entonces (a) existe un tiempo finito k^* después de que para todo $k > k^*$, $J_r(k)$ está acotada en $[1, K]$, $K > k^*$, (b) el proceso de conmutación de los modelos neuronales paran, i.e., r es constante.

5. Resultados

La planta a identificar es

$$y(k) = \frac{y(k-1)y(k-2)[y(k-1)+2.5]}{(1+y(k-1))^2+y(k-2)^2} + u(k-1) \quad (14)$$

La señal de entrada entrenada es seleccionada como números aleatorios en el intervalo $[0,1]$.

$$\begin{aligned} X(k) &= [y(k-1), y(k-2), u(k-1)]^T \\ &= [x_1(k), x_2(k), x_3(k)]^T \end{aligned}$$

Se selecciona $\alpha=0.4$. Como conocido a priori, se conocen los cambios máximos en la entrada y la salida son alrededor de 3 y 1, $\alpha \|x_{max} - x_{min}\| + (1-\alpha) \|y_{max} - y_{min}\| = 1.8\alpha$. Así L podría ser escogida tal que $L < 1.8$, en esta aplicación se selecciona $L=1.5$

El resultado utilizando el enfoque multimodelo y el intercambio de cada grupo se muestra en la figura 3.

6. Conclusión

En este artículo se propuso un nuevo kernel para la estructura de los support vector machines estándar, y se observó que en el proceso de extracción de reglas difusas para

el modelado de sistemas no lineales es llamado estructura de identificación. A la combinación de multimodelo y redes neuronales difusas se les considera como un enfoque racional para identificación de sistemas no lineales. Se muestra un rango de aprendizaje variante en el tiempo para uso común del algoritmo retro-propagación (*backpropagation*), se prueba que los errores de identificación están acotados, el cual cambiará el modelo de la red neuro difusa solo cuando las funciones de membresía son casi iguales.

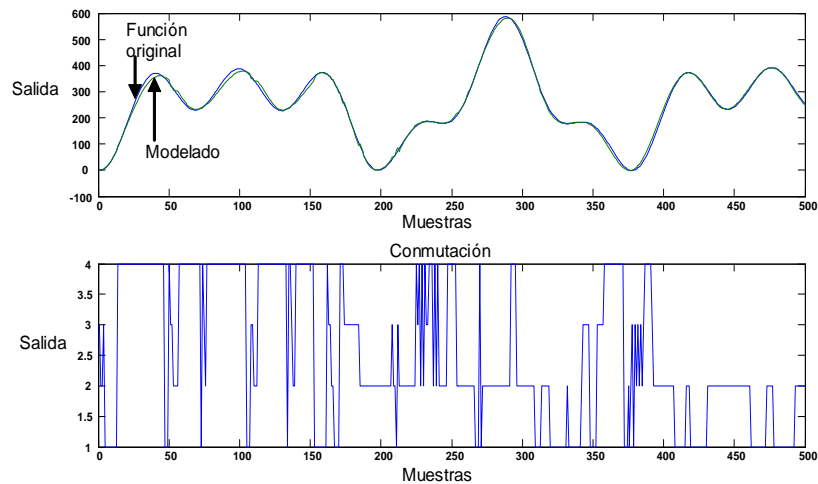


Fig. 3. Enfoque multimodelo del sistema no lineal

Referencias

1. Wang, L.X.: Adaptive Fuzzy Systems and Control. Englewood Cliffs NJ: Prentice-Hall (1994)
2. Leski, Jacek M.: TSK-Fuzzy Modeling Based on ϵ -Insensitive Learning. IEEE Trans. on Fuzzy System, vol. 13, no. 2, pp181-193 (2005)
3. Rivals I. and Personnaz, L.: Neural-network construction and selection in nonlinear modeling. IEEE Transactions on Neural Networks, Vol.14, No.4, 804-820 (2003)
4. Tovar, J., Wen Yu, Ortiz, F., Román Mariaca, C., and Rubio, J.J.: Modeling via on-line clustering and fuzzy support vector machines for nonlinear system. In: 50th IEEE Conference on Decision and Control and European Control Conference, December 12-15, Orlando, FL, USA (2011)

Mejorando el clasificador asociativo de patrones en bases de datos multi-clase

Roberto O. Cruz L., Raúl Santiago M. y Héctor J. Puga S.

Instituto Tecnológico de León, Av. Tecnológico, Industrial Julián de Obregón,
León, Guanajuato, 37000, México
{robanac,rsantiago66}@gmail.com,pugahector@yahoo.com

Resumen En el siguiente artículo se muestra la modificación del clasificador supervisado Clasificador Asociativo de Patrones (CAP) donde se implementa una modificación al proceso de traslación de ejes obteniendo una clasificación en cascada. Se pone a prueba la modificación y se clasifican distintas bases de datos multi-clase observando su comportamiento en distintos escenarios. En la interpretación de resultados se utiliza el método de validación K-fold y la prueba estadística no paramétrica de Wilcoxon, reforzando los resultados obtenidos. Con la modificación al algoritmo CAP se obtiene una mejora sustancial así como mejores rendimientos de clasificación.

Palabras clave. Memorias asociativas, rendimiento de clasificación, CAP, cascada.

1. Introducción

El área de Reconocimiento de Patrones es un campo donde existe una amplia variedad de técnicas que compiten entre sí por ofrecer los mejores resultados. Entre ellas las Redes Neuronales, Maquinas de Vector Soporte, Knn, Bayesiano y Mínima Distancia, por citar algunas. En este campo es donde nuestra propuesta de investigación se desenvuelve y utiliza el modelo de Memorias Asociativas en procesos de clasificación.

En [1] mencionan lo siguiente sobre las Memorias Asociativas aplicadas a la clasificación de patrones: *“In acts of associative memory, the system takes in a pattern and emits another patten which is representative of a general group of patterns. It thus reduces the information somewhat, but rarely to the extent that pattern classification does”*. Es entonces que las Memorias Asociativas son solo un sistema que toma un patrón y emite otro, el cual es generalmente representativo de un grupo de patrones ocasionando una reducción de la información, pero raramente en la medida de clasificar patrones.

En el año del 2002 [2] propone el primer clasificador supervisado basado en Memorias Asociativas llamado Clasificador Híbrido Asociativo con Traslación (CHAT) y mostró resultados promisorios de eficacia, compitiendo con otros clasificadores en el ámbito de clasificación de patrones más longevos. En el año 2003 los creadores de CHAT lo renombraron por Clasificador Asociativo de Patrones (CAP) [3].

En las pruebas desarrolladas en [4] se muestran rendimientos de clasificación mayores al 90% usando bases de datos (BD) bi-clase y se plasma un procedimiento multi-clase elitista de selección de patrones. Este procedimiento elitista no es común en procesos reales de clasificación y bajo escenarios no controlados de selección de los patrones.

En este trabajo se propone una extensión al CAP para aplicarlo a problemas de clasificación multi-clase. Esta extensión se ha denominado como *Técnica en Cascada*. La propuesta mejora el rendimiento de clasificación en BD multi-clase con respecto al modelo original del CAP y evita cualquier pre-procesamiento de los datos. La propuesta muestra un mejor rendimiento de clasificación sobre BD ampliamente utilizadas en pruebas de clasificación como la BD Iris Plant. Además se establece un marco comparativo con la versión original del CAP aplicando técnicas estadísticas.

Este artículo está organizado de la manera siguiente: En la sección 2 se muestra el funcionamiento general del CAP; En la sección 3 se muestra el diseño de la técnica Cascada y los elementos estructurales que la describen; En la sección 4 se pone a prueba la técnica en cascada usando distintas BD, se implementan técnicas de validación y selección de datos como *K-Fold Validation* y la prueba estadística no paramétrica de *Wilcoxon* para determinar qué algoritmo es mejor; y para finalizar, en la sección 5 se muestran algunas conclusiones.

2. CAP

Para comprender la mejora del algoritmo CAP para BD multi-clase con la técnica cascada, primero se ilustrará el funcionamiento del CAP y las ventajas que presenta este algoritmo. En [4] se consideran como principales características del clasificador los siguientes puntos:

- El CAP combina los principios de álgebra de matrices que utiliza el Linear Associator [5] y el criterio de clasificación de la Lernmatrix [6];
- Siendo una Memoria Asociativa que funciona como clasificador cuenta con las fases de aprendizaje y de recuperación;
- Para la fase de aprendizaje y posteriormente para la fase de recuperación se realiza una traslación de todos los patrones del conjunto fundamental empleando un vector medio, generado a partir de un conjunto fundamental;
- La posición del vector medio da origen a un nuevo conjunto de ejes coordenados;
- En la fase de recuperación, si se desea clasificar un patrón desconocido que no exista dentro del conjunto fundamental, es necesario trasladar al nuevo espacio de clasificación definido a través del vector medio.

En forma general, el algoritmo del CAP es el siguiente:

1. Sea un conjunto fundamental de patrones de entrada de dimensión n con valores reales en sus componentes (a la manera del Linear Associator), que se aglutinan en m clases diferentes.
2. A cada un de los patrones de entrada que pertenece a la clase k se le asigna el vector formado por ceros, excepto en la coordenada k -ésima, donde el valor es uno (a la manera de la Lernmatrix).

3. Se calcula el vector medio del conjunto de patrones muestra o prototipos con que se cuenta.
4. Se toman las coordenadas del vector medio a manera de centro de un nuevo conjunto de ejes coordenados.
5. Se realiza la traslación de todos los patrones del conjunto fundamental.
6. Aplicamos la fase de aprendizaje, que es similar a la del Linear Associator.
7. Aplicamos la fase de recuperación, que es similar a la que usa la Lernmatrix.
8. Trasladamos todo patrón a clasificar en los nuevos ejes.
9. Procedemos a clasificar los patrones desconocidos.

En la Figura 1 se muestra en a) una representación de dos clases y el vector medio que se genera bajo la propuesta del CAP. Este vector dará origen al nuevo espacio de clasificación, además se ilustra en b) el nuevo espacio de clasificación y los patrones trasladados con respecto al vector medio. Se observa como la traslación garantiza una separación entre las clases, intuitivamente es un efecto exclusivo para clasificación bi-clase.

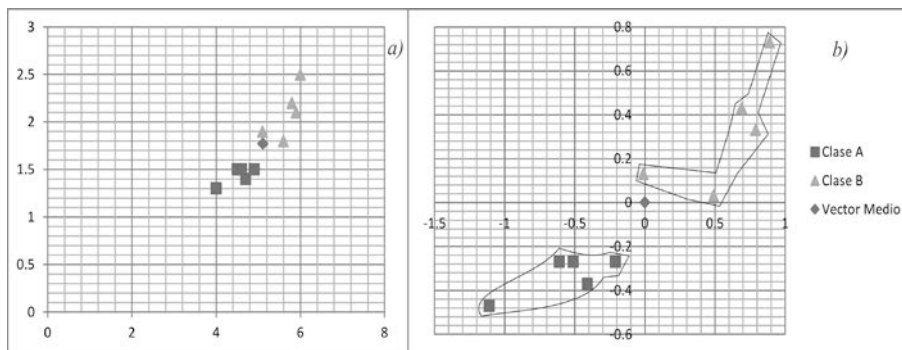


Fig. 1. Patrones simulados en dos clases y la traslación de los mismos (Fuente: Elaboración propia)

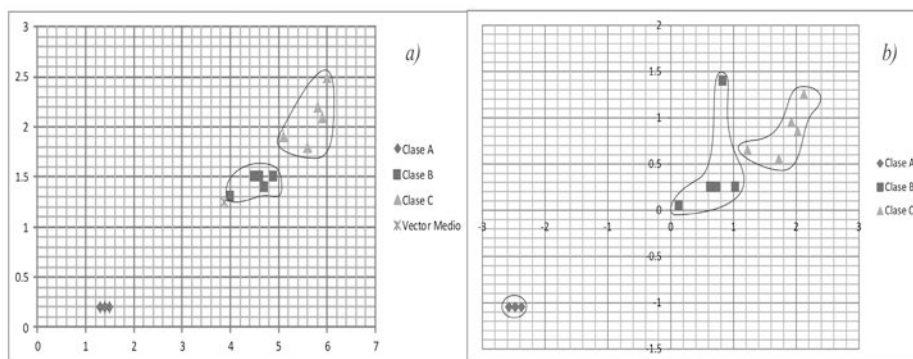


Fig. 2. Simulación de patrones en tres clases y la traslación de los mismos (Fuente: Elaboración propia)

En la Figura 2 se muestra en a) una representación de tres clases y el vector medio que se genera bajo la propuesta del CAP. Este vector dará origen al nuevo espacio de clasificación, además se ilustra en b) el nuevo espacio de clasificación y los patrones trasladados con respecto al nuevo origen.

Como puede observarse en el ejemplo de la Figura 2 para un problema multi-clase el CAP agrupa dos clases como si fuese una sola donde el vector medio no puede garantizar una separabilidad entre las clases. De forma general el CAP tiende a obtener bajos rendimientos en procesos de clasificación multi-clase.

3. Técnica cascada

Nuestra propuesta consiste en transformar un proceso multi-clase en múltiples procesos bi-clase. Los procesos de clasificación bi-clase dependen del total de clases con las que se cuente en la BD. La modificación al CAP tiene las siguientes consideraciones:

1. El número de niveles define la cantidad de bloques en los que se desarrolla el proceso de clasificación.
2. En cada bloque se obtiene un vector medio, que dará origen a un nuevo espacio de clasificación.
3. Como los vectores medios van cambiando, la matriz de entrenamiento tiene que ser calculada en cada bloque o nivel.
4. En cada bloque se obtiene un resultado de clasificación para una clase de la base de datos.

Por ejemplo, si una BD cuenta con cuatro clases el proceso de clasificación se desarrolla en cuatro etapas, cada etapa se encarga de clasificar todos los elementos que describen a una clase en particular, después de clasificar los primeros patrones se continúa con los restantes y así sucesivamente hasta terminar la clasificación de todas las clases involucradas en el planteamiento inicial. En la Figura 3 se muestra un esquema de nuestro planteamiento.

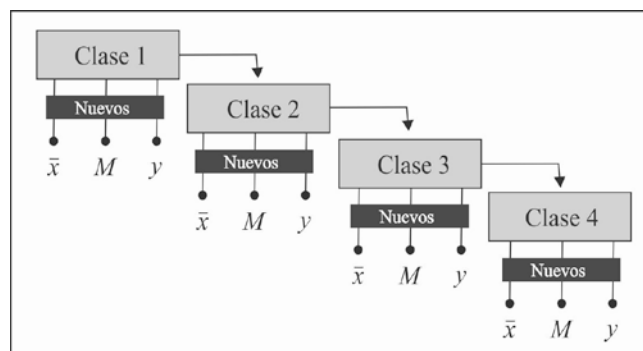


Fig. 3. Representación de Niveles de la Cascada donde \bar{x} representa cada vector medio, M representa la nueva matriz de entrenamiento y y el resultado de clasificación (Fuente: Elaboración propia)

Ya se menciona anteriormente que al obtener un vector medio en cada nivel o bloque se genera un nuevo proceso de clasificación bi-clase. En la siguiente sección se describe cómo se generan los vectores medios de nuestra propuesta.

3.1. Obtención de los vectores medios (VM)

El vector promedio de una clase se puede visualizar como un elemento representativo de dicho conjunto de vectores. Si tenemos m vectores que representan a una clase A el vector promedio \bar{x} se define de la siguiente manera

$$\bar{x} = \frac{\sum_{i=1}^m v_i}{m} \quad (1)$$

Siendo v_i un patrón en la clase A .

Bajo la propuesta del CAP, el vector medio obtenido genera un nuevo espacio de clasificación. Ahora:

Suponiendo que se tienen tres clases distintas A, B y C , el vector medio de estas no puede garantizar una separabilidad entre las clases. La propuesta de cascada genera artificialmente dos clases, una de ellas es la clase que se desea separar y la otra la aglutinación de las clases restantes. Si se desean clasificar los patrones correspondientes a la clase A , es necesario conocer un origen que de lugar a un espacio de clasificación. El nuevo origen del espacio de clasificación se realiza a través de la unión de las clases B y C (clases restantes) como una sola. A esta unión se le ha denominado *clase nube* y la clase que se está clasificando se le denomina *clase principal*. Después se calcula el vector promedio de la *clase nube*, ya no es necesario calcular el promedio con todos los vectores ya que es suficiente con obtener el promedio de los vectores medios \bar{x}_2 , de la clase B y \bar{x}_3 , de la clase C . Esto da origen a un vector medio del bloque que hemos definido como \bar{n} .

En la Figura 4 se muestra la *clase principal* A junto a su vector promedio \bar{x}_1 , además se muestra una representación de la *clase nube* BC y el vector promedio de la nube.

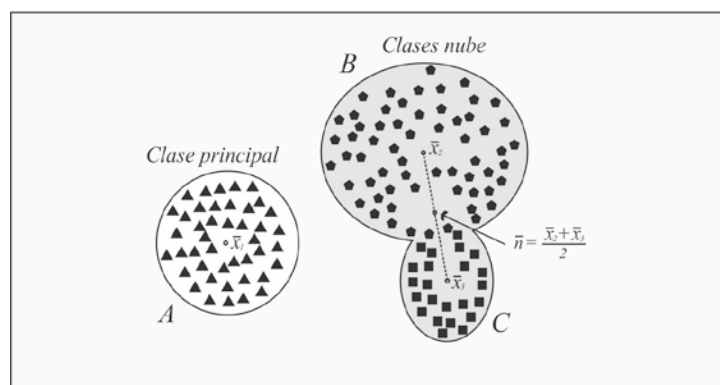


Fig. 4. Ejemplo de Clase principal y Clases nube, vectores promedio $\bar{x}_1, \bar{x}_2, \bar{x}_3$ y \bar{n} (Fuente: Elaboración propia)

Una vez obtenidos el vector promedio de la *clase principal* que en este caso es \bar{x}_1 y el vector promedio de las *clases nube* \bar{n} , el nuevo origen del conjunto de ejes coordenados o *vector medio* queda definido como

$$VM = \frac{\bar{x}_1 + \bar{n}}{2} \quad (2)$$

Como se puede ver en la ecuación 2 solo es necesario calcular el vector medio VM entre el vector promedio de la *clase principal* y el vector promedio de las *clases nube*.

En la Figura 5 se muestra el VM cuando la clasificación se realiza con la clase A.

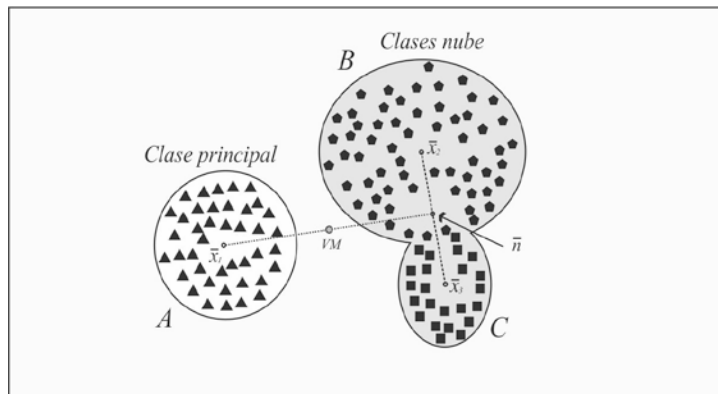


Fig. 5. Vector Medio en clasificación de clase A (Fuente: Elaboración propia)

Es fundamental señalar que todo el procedimiento de obtención de VM se desarrolla por clase, es decir, en su momento todas ellas llegan a ser clase principal y las restantes clases nube, obteniendo en cada bloque un nuevo VM .

3.2. Algoritmo CAP cascada

Algoritmo General de CAP Cascada

Se detalla el algoritmo general del CAP Cascada:

- Sea un conjunto de n clases distintas con m vectores cada una;
- Sea un conjunto fundamental de patrones de entrada de dimensión j con valores reales en sus componentes (a la manera del Linear Associator), que se aglutinan en n clases distintas;
- A cada uno de los patrones de entrada que pertenece a la clase k se le asigna el vector formado por ceros, excepto en la coordenada k -ésima, donde el valor es uno (a la manera de la Lernmatrix);
- Se calculan los vectores promedio de cada clase;
- Se implementa n veces el *algoritmo de clasificación por clase*;

Se unen los resultados de clasificación.

Algoritmo de Clasificación por Clase

Este procedimiento se realiza por cada clase:

- Se calcula el *vector medio* con respecto al vector promedio de la *clase primaria* y al vector promedio de las *clases nube*;
- Se toman las coordenadas del *vector medio* a manera de centro de un nuevo conjunto de ejes coordenados;
- Se realiza la traslación de todos los patrones del conjunto fundamental;
- Aplicamos la fase de aprendizaje, que es similar a la del Linear Associator;
- Aplicamos la fase de recuperación, que es similar a la que usa la Lernmatrix;
- Trasladamos todo patrón desconocido de la *clase primaria* a clasificar en los nuevos ejes;
- Procedemos a clasificar los patrones desconocidos;
- Se almacenan los resultados de clasificación.

4. Resultados

En esta sección es necesario mencionar cuáles son las bases de datos con las que se está poniendo a prueba el CAP y la propuesta CAP Cascada. Todas ellas se obtuvieron del UCI Machine Learning Repository de la universidad de California [7], este repositorio cuenta con alrededor de 1000 citas lo cual otorga a la fuente alta confiabilidad. Las bases de datos utilizadas son las siguientes:

Table 1. Características de las Bases de Datos.

BD	Instancias	Número Atributos	Clases	Año
Iris	150	4	3	1988
Wine	178	13	3	1991
Glass	214	9	6	1987
Statlog Project S. Vehículos	846	18	4	1986
Segmentation	210	19	7	1990

Las pruebas se desarrollaron en una PC genérica con procesador Intel Core 2 Quad a 2.40Ghz, 4 Gb de Ram, disco duro de 145 Gb, sistema operativo Windows 7 de Microsoft y la programación se realizó en lenguaje Java.

4.1. Comparativo entre CAP y CAP cascada

En esta sección se muestran los rendimientos de clasificación que se obtienen del CAP y CAP Cascada en cada una de las bases de datos descritas anteriormente, ambos clasificadores son sometidos en un mismo escenario de clasificación donde se implementa una validación cruzada de K-Fold [8] usando $K = 10$, este proceso se realiza 100 veces obteniendo un rendimiento promedio de cada clasificador en cada una de las bases de datos. Para determinar que clasificador obtiene mejor rendimiento

se implementa la prueba estadística no paramétrica de Wilcoxon con todos los promedios de rendimiento obtenidos.

En la Tabla 2 se muestran los porcentajes de rendimientos promedio de clasificación del CAP y CAP Cascada en cada una de las bases de datos

Table 2. Porcentajes de Rendimiento promedio de clasificación, usando K-Fold con $K = 10$ en 100 épocas de clasificación.

BD	CAP	CAP Cascada	Aumento de Porcentaje
Iris Plant	69.04	79.40	10.35
Wine	68.18	76.27	8.09
Glass	50.04	81.04	31.00
Statlog Project S. Vehículos	38.52	62.08	23.55
Segmentation	62.21	73.97	11.75

En la Figura 6 se muestran gráficas de dos de los comportamientos de clasificación de la Tabla 2, donde las líneas rojas corresponden al CAP y las líneas azules al CAP Cascada

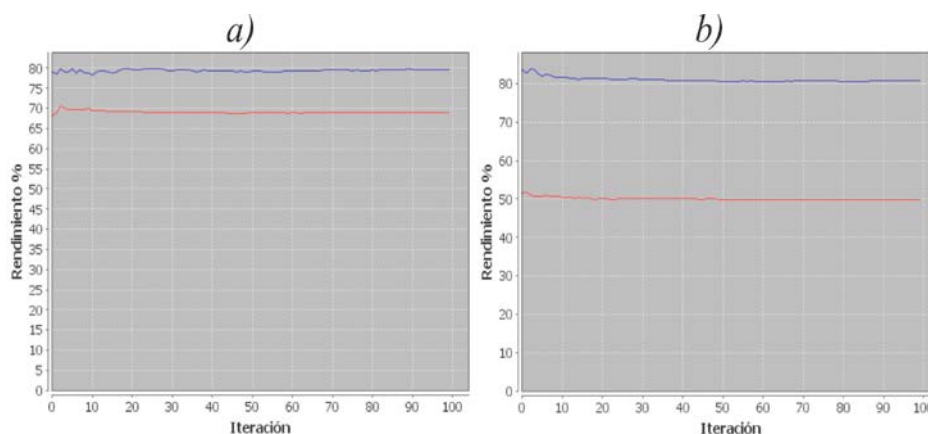


Fig. 6. a) Gráfica de rendimientos de clasificación en BD Iris Plant, b) Gráfica de rendimientos en BD Glass (Fuente: Elaboración propia)

4.2. Prueba Wilcoxon

La prueba estadística de Wilcoxon [9] es una herramienta matemática la cual determina si dos distribuciones son iguales, es considerada como una prueba no paramétrica ya que analiza experimentos que producen mediciones de respuesta que son difíciles de interpretar o cuantificar. Este método es bueno para resolver suposiciones generales acerca de las características de cualquier distribución de probabilidad o parámetros involucrados en un problema inferencial.

Esta prueba ayuda a determinar qué clasificador es *mejor*, para implementar la prueba es necesario definir una Hipótesis nula H_0 , la cual en todos los casos refiere a que las dos o más distribuciones comparadas son iguales. Para este ejemplo se espera que cuando la distribución de CAP y la distribución de CAP Cascada sean comparadas exista una diferencia notable entre ellas. De ser así, la hipótesis H_0 estaría siendo rechazada.

Para determinar qué clasificador es mejor se toman en cuenta los siguientes puntos:

- Es necesario tener como mínimo 5 instancias de prueba para la prueba básica;
- Iniciamos los parámetros $\alpha = 0.1$ y $T_{critico} = 1$ (parámetros definidos por Wilcoxon para un prueba de cinco instancias)[9];
- Los escenarios de clasificación deben ser los mismos, siendo los patrones de entrenamiento y de clasificación los mismos para CAP como para CAP Cascada, es por eso que utilizamos los datos obtenidos en la validación K-Fold mostrados en la Tabla 2;
- Definimos dos distribuciones $distX$ y $distY$ las cuales pertenecen a CAP y CAP Cascada respectivamente.

Los resultados de la prueba Wilcoxon son los siguientes:

Table 3. Cálculos de Wilcoxon con rendimientos de clasificación de CAP y CAP Cascada

BD	$distX$	$distY$	D	$D_{Absoluta}$	Rango
Iris Plant	69.04	79.40	-10.35	10.35	2
Wine	68.18	76.27	-8.09	8.09	1
Glass	50.04	81.04	-31.00	31.00	5
Statlog Project S. Vehículos	38.52	62.08	-23.55	23.55	4
Segmentation	62.21	73.97	-11.75	11.75	3

Después de los resultados mostrados en la Tabla 3 $T- = 15$ y $T+ = 0$, utilizando la literatura [9] se concluye que $T+ < T_{critico}$, por lo tanto la hipótesis nula H_0 se rechaza, y sabemos que la distribución $distY$ (CAP Cascada) se encuentra más hacia la derecha teniendo los valores más altos de ambas distribuciones y por lo tanto es mejor.

5. Conclusiones

Los resultados experimentales obtenidos en este trabajo muestran que el CAP tiene un menor rendimiento en clasificación multi-clase en comparación con procesos bi-clase, esto limita al algoritmo de clasificación para ser aplicado de manera general.

Nuestra propuesta de clasificación en cascada mejora sustancialmente al modelo original del CAP sin tener que utilizar un pre-procesamiento de los datos y siguiendo el mismo modelo. Finalmente la prueba de Wilcoxon valida que en el escenario de clasificación que se usa en este artículo, la propuesta cascada es superior al algoritmo CAP original.

Referencias

1. Duda,R.O.,Hart, P.E. & Stork,D.G.: Pattern Classification.Wiley,USA (2001)
2. Santiago Montero, R., Díaz-de-León Santiago, J.L. & Yáñez Márquez, C.: Clasificador híbrido de patrones basado en la Lernmatrix de Steinbuch y el Linear Associator de Anderson-Kohonen. In Díaz-de-León Santiago, J.L. & Yáñez Márquez, C. (Eds.) "Reconocimiento de Patrones. Avances y Perspectivas", Special Edition,Vol. 1, ISBN 970-36-0194-4, CIC-IPN, México, pp. 449-460.Research on Computing Science Series (2002)
3. Santiago Montero, R., Díaz-de-León Santiago, J.L. & Yáñez Márquez, C.: Clasificador asociativo de patrones: avances teóricos. In Díaz de León-Santiago, J.L., González-Santos, G. & Figueroa-Nazuno, J. (Eds.) "Avances en: Ciencias de la Computación", Special Edition, Vol. 3, ISBN 970-36-0194-4, CIC-IPN, México, pp. 257-267.Research on Computing Science Series (2003)
4. Santiago Montero, R.: Clasificador Híbrido de Patrones basado en la Lernmatrix de Steinbuch y el linear Associator de Anderson-Kohonen. Tesis de Maestría, Instituto Politécnico Nacional-Centro de Investigación en Computación (2003)
5. Yáñez, C., Díaz de León, J.L.: Linear Associator de Anderson-Kohonen. Colección Memorias Asociativas, Instituto Politécnico Nacional-Centro de Investigación en Computación (2001)
6. Yáñez, C., Díaz de León, J.L.: Lernmatrix de Steinbuch. Colección Memorias Asociativas, Instituto Politécnico Nacional-Centro de Investigación en Computación (2001)
7. Frank, A. & Asuncion, A.: UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>]. Irvine, CA: University of California, School of Information and Computer Science (2010)
8. Kohavi, Ron.: A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. Article, Computer Science Department-Stanford University (1995)
9. Mendenhall, W.: Estadística Matemática con Aplicaciones. Iberoamericana (1986)

Análisis probabilístico mediante redes bayesianas para el cálculo de la probabilidad de lluvia en diferentes localidades

Guillermo De la Torre-Gea y Oscar Delfín-Santiesteban

Tecnológica de Corregidora, Municipio de Corregidora, Santiago de Querétaro, México
gtorre.utcorregidora@gmail.com

Resumen La comprensión de las relaciones entre los factores climáticos contribuye a un mejor conocimiento de los fenómenos atmosféricos. Dichos factores se estudian mediante el diseño de modelos predictivos, sin embargo, las técnicas estadísticas convencionales no toman en cuenta las relaciones de dependencia entre estos factores. Los modelos de aprendizaje bayesianos consisten en la cuantificación de la probabilidad condicional, dando como resultado la identificación de relaciones causales entre las variables. En este trabajo se empleó un modelo de redes bayesianas para realizar el análisis probabilístico que nos permitió determinar las dependencias espaciales y temporales entre las precipitaciones de diferentes localidades que no son observables con otros métodos. Tomando en cuenta un conjunto de datos incompletos de la precipitación pluvial en 18 estaciones meteorológicas a partir de tres años, las relaciones de dependencia entre dichos valores de precipitación se pueden observar y calcular las probabilidades de lluvia condicionales.

Palabras clave: Modelos probabilísticos, clima, predicción, minería de datos, K2.

1. Introducción

La climatología se basa en un análisis estadístico de la información meteorológica medida y almacenada, las variaciones temporales que se producen en los parámetros climáticos se incorporan a los promedios estadísticos [14]. De esta forma, los factores que conforman el clima se estudian para desarrollar modelos de predicción. Para este propósito, es necesario obtener conjuntos de datos mediante un método sistémico y homogéneo, a partir de las estaciones meteorológicas durante períodos por lo menos de 30 años para ser considerados representativos. Las bases de datos climáticas contienen propiedades estadísticas sobre la precipitación pluvial para cada localidad, mientras que análisis conjunto de estas bases de datos podría contener las relaciones entre las precipitaciones de diferentes localidades. Así, esta información puede ser utilizada para analizar los problemas relacionados con la dinámica de la atmósfera y los posibles efectos que pueden causar en la climatología de las regiones. Sin embargo, las técnicas estadísticas en estos problemas, incluidos los métodos de regresión lineal para el pronóstico del tiempo, los métodos de agrupamiento y análisis de componentes principales para la identificación de los patrones atmosféricos de representación, contienen información fragmentada y asumen independencias espaciales ad-hoc

con el fin de obtener modelos simplificados [4], [2]. Por otro lado, los datos sobre precipitación pluvial obtenidos de las estaciones climáticas son a menudo incompletos, lo que obliga al uso de técnicas de minería de datos para el análisis [1], [11]. Por estas razones, es necesario aplicar técnicas de análisis que no afecten las propiedades inherentes de dichos datos.

Los modelos numéricos de predicción del clima son representaciones abstractas del mundo real. Estos modelos discretizan los datos utilizando funciones de aproximación para describir el comportamiento de las variables climáticas de interés en los estudios de predicción.

Hoy en día, los modelos numéricos son indispensables para la predicción climática. De acuerdo con [15], los parámetros en escala de tiempo variable podrían ser estimados utilizados para evaluar si existen tendencias estadísticamente significativas en datos climáticos a través de redes bayesianas (RB). Por otra parte, [21] propuso un modelo estocástico del espacio-tiempo que representa las dependencias temporales y espaciales de ocurrencia de precipitación por día. Un modelo RB para estimar la ocurrencia de precipitación basado en cadenas ocultas de Markov fue desarrollado por [10] utilizando el método de estimación de máxima probabilidad con datos incompletos. Según [13], es recomendable el empleo de RB sobre todo en conjuntos pequeños de datos disponibles.

El objetivo de este estudio es demostrar que las RB pueden ser empleadas para encontrar un modelo que mejor describa las relaciones entre la presencia de precipitación pluvial en distintas estaciones meteorológicas, mediante datos disponibles incompletos y limitados en el tiempo, con lo cual realizar el pronóstico de precipitación calculando las probabilidades condicionales de ocurrencia de precipitación pluvial.

2. Teoría de las RB

Las RB son representaciones del conocimiento desarrollados en el campo de la inteligencia artificial para el razonamiento aproximado [18], [16] y [5]. Una RB es un gráfico acíclico cuyos nodos corresponden a conceptos o variables, y cuyos enlaces definen las relaciones o funciones entre dichas variables [3]. Las variables se definen en un dominio discreto o cualitativo, y las relaciones funcionales describen las inferencias causales expresadas en términos de probabilidades condicionales, como se muestra en la Ecuación 1.

$$P(x_1, \dots, x_n) = \prod_{i=1}^n P(x_i | \text{padres}(x_i)) \quad (1)$$

Las RB pueden ser usadas para identificar las relaciones entre las variables anteriormente indeterminadas o para describir y cuantificar estas relaciones, incluso con un conjunto de datos incompletos [11] y [19]. Los algoritmos de solución de las RB permiten el cálculo de la distribución de probabilidad esperada de las variables de salida. El resultado de este cálculo es dependiente de la distribución de probabilidad de las variables de entrada. A nivel mundial, las RB puede ser percibida como una

distribución de probabilidades conjunta de una colección de variables aleatorias discretas [7].

La probabilidad a priori $P(c_j)$ es la probabilidad de que una muestra x_i pertenezca a la clase C_j , definida sin ninguna información sobre sus valores característicos, como se muestra en la Ecuación 2.

$$P(c_j / x_i) = P(x_i / c_j)P(c_j) / \sum P(x_i / c_k)P(c_k) \quad (2)$$

Las máquinas de aprendizaje, en la inteligencia artificial, está estrechamente relacionado con la minería de datos, métodos de clasificación o agrupamiento en estadística, razonamiento inductivo y reconocimiento de patrones. Métodos estadísticos de aprendizaje automático se pueden aplicar al marco de la estadística bayesiana, sin embargo, la máquina de aprendizaje pueden emplear una variedad de técnicas de clasificación para producir otros modelos de RB [17] y [20]. El objetivo del aprendizaje mediante RB es encontrar el arreglo que mejor describa a los datos observados. El número de posibles estructuras de grafos acíclicos directos para la búsqueda es exponencial al número de variables en el dominio, el cual se encuentra definido por la Ecuación 3:

$$f(n) = \sum_{i=1}^n (-1)^{i+1} C^n 2_i^{(n-i)} f(n-i) \quad (3)$$

El algoritmo K2 constituye el método más representativo entre las aproximaciones de “búsqueda y resultado”. El algoritmo comienza asignando a cada variable sin padres. A continuación, agrega de manera incremental los padres a la variable actual que aumenta su puntuación en la estructura resultante. Cuando cualquier adición de una madre soltera no puede aumentar la cuenta, deja de agregar padres a la variable. Desde un pedido de las variables conocido de antemano, el espacio de búsqueda bajo esta restricción es mucho menor que el espacio tomando la estructura entera, y no hay necesidad de comprobar los ciclos en el proceso de aprendizaje. Si el orden de las variables es desconocida, se puede realizar la búsqueda en los ordenamientos [8].

3. Materiales y métodos

El área de estudio abarca el estado de Querétaro, México, que tiene una superficie de 11.689 km² divididos en tres zonas climáticas diferentes: la parte sur, que cubre la provincia fisiográfica del Eje Neovolcánico, la región central que comprende las áreas de la región del Eje Neovolcánico, la Sierra Madre Oriental y la Mesa Central y la zona norte que corresponde a una porción de la Sierra Madre Oriental, como se muestra en la Fig. 1.

Se eligieron 18 estaciones climáticas en el estado de Querétaro para obtener los datos de precipitación pluvial, que representan los tres climas diferentes (Fig. 2 y 3), ubicando a Jalpan que corresponde a un clima de ACw, las estaciones meteorológicas que corresponde a un clima BS1k son: Cadereyta, Ezequiel Montes, Colón y Tequisquiapan, mientras que Amealco, San Joaquin y Huimilpan corresponden a un clima Cw de acuerdo a la clasificación de Koppen [6], como se muestra en la Fig. 2. Las



Fig. 1. Fisiografía del estado de Querétaro, fuente [12].



Fig. 2. Distribución climática del estado de Querétaro, fuente [12].

estaciones meteorológicas Corregidora, El Marques, CEA, El Milagro, Centro Sur, Centro Cívico, Rochera y Altamira corresponden a climas de transición.

Se obtuvieron datos de cada día durante el período comprendido entre el 01/01/2007 al 31/12/2010, a través de la red de estaciones meteorológicas de la Comisión Estatal de Aguas del estado de Querétaro, cuya ubicación se presenta en la Fig. 3, mediante el portal:

<http://www.wunderground.com/weatherstation/ListStations.asp?selectedCountry=Mexico>.

El conjunto de datos estaba compuesto por valores máximos, promedio y mínimos de precipitación pluvial, los cuales fueron discretizados en 2 valores: presencia y ausencia de lluvia.



Fig. 3. Estaciones meteorológicas en el estado de Querétaro, fuente [12].

3.1. Análisis bayesiano

El análisis fue realizado mediante el sistema ELVIRA versión 0.162 en tres etapas sugeridas por [16]:

1. Pre-proceso y tratamiento de datos: Se empleó el algoritmo de imputación por promedios, el cual sustituye los valores perdidos/desconocidos con la media de cada variable. No necesita parámetros. La discretización de las variables fue de tipo masiva, empleando el algoritmo Equal frequency con número de intervalos igual a dos (presencia y ausencia de precipitación pluvial).
2. Aprendizaje automático: Se realizó mediante el método de aprendizaje K2 learning, con máxima verosimilitud y número máximo de padres igual a 5, sin restricciones.
3. Post-proceso:
 - a) Formulación del esquema de dependencias entre variables: A partir del conjunto de datos de las 18 estaciones climatológicas, se realizó un análisis de dependencia entre todas las variables. Este análisis consiste en la cuantificación de los diferentes tipos de dependencia y ha dado como resultado la identificación de las relaciones causales existentes entre las variables.
 - b) Aprendizaje estructural de las redes Bayesianas: A partir de la identificación de las relaciones causales entre variables, se han determinado las estructuras

de redes Bayesianas que mejor describen el comportamiento de la precipitación pluvial en las diferentes localidades del estado de Querétaro.

- c) Aprendizaje paramétrico de la RB: Una vez obtenida la estructura topológica de la RB, se han obtenido los parámetros o distribuciones de probabilidades condicionadas entre variables que permiten representar cada uno de los arcos que componen las estructuras de RB. Estos parámetros permiten obtener las probabilidades de ocurrencia de precipitación pluvial para cualquiera de las variables que componen la red [16].

4. Resultados y discusión

Se obtuvo un modelo de RB a partir de los datos de las 18 estaciones meteorológicas del estado de Querétaro, el cual se muestra en la Fig. 4.



Fig. 4. Modelo de RB que muestra las relaciones entre las estaciones meteorológicas.

Se observa la interacción de dos localidades: Amealco y C. Sur que se están presentes con CEA, las cuales se encuentran distantes unas de otras, lo cual sugiere un sistema pluvial con alta probabilidad con dirección Sur a Norte incluso interactuando con Sta. Rosa y Altamirano que se encuentran a mayor distancia y que no incluyen otras localidades más cercanas. Otro sistema pluvial indica la interacción entre localidades orientadas de Oeste a Este como son: El Milagro, C. Cívico y C. Sur con Tequisquiapan y San Juan del Río las cuales se encuentran a mayor distancia. Y un tercer sistema pluvial con dirección Oeste a Noreste que involucran las localidades El Milagro, Corregidora con Colón que se encuentra a mayor distancia y que no incluyen localidades en puntos intermedios, quizás debido a la topografía de la región.

Las interacciones entre localidades implican la presencia de lluvia en estos sitios y aunque no necesariamente al mismo tiempo, debe ser en espacios de tiempo breve. Por lo que dichas interacciones pueden estar relacionadas con los vientos dominantes, los cuales tienen un efecto de movilidad sobre la precipitación pluvial. Por otra parte, la dirección de las interacciones es de localidades con climas húmedos hacia aquellas que tienen clima más seco.

Table 1. Probabilidades condicionales de las principales relaciones inferidas a partir de la RB obtenida.

CEA .77	Colon .875	Sta Rsa .9	C.Cívico .82	Altamira .88	Rochera .84	El Marques .99	SJR .99	Tequis- quiapan .99
C.Sur	Huimilpan	Amealco	Amealco	Amealco	El Milagro	Huimilpan	El Milagro	El Milagro
Amealco	Milagro	CEA	Huimilpan	C.Sur	Corregidora	El Milagro	C.Cívico	C.Cívico
	Corregidora			CEA	C.Cívico	C.Cívico	El Marquez	C.Sur
						Jalpan	Colon	
							E.Montes	

Las interacciones que involucran pocas localidades tienen valores de probabilidad menores, las cuales se muestran en la Tabla 1, que aquellas en las que el número de localidades es mayor y se encuentran más distantes entre sí, lo que sugiere estar relacionado con fenómenos meteorológicos locales, en el primer caso y generalizados para el segundo caso. Esto se corrobora tomando en cuenta que las interacciones con mayor número de localidades y mayor probabilidad, también son aquellas que presentan un clima más seco, en donde la presencia de precipitación pluvial se da en eventos generalizados y no locales, bajo la influencia de otras localidades con climas más húmedos.

Por otra parte, se presentan localidades cercanas que muestran independencia en la precipitación pluvial, lo cual puede ser un factor de interés relacionado con la presencia de corrientes de aire que no permiten que la precipitación sea homogénea, a pesar de que la topografía de la región sea regular.

Para los alcances de este trabajo se cubierto con el objetivo, sin embargo, es necesario realizar un estudio más detallado tomando en cuenta otros métodos de *machine learning* que permitan realizar una comparación.

5. Conclusiones

Es posible determinar las relaciones de dependencia causal entre los valores de ocurrencia de precipitación pluvial en diferentes localidades mediante el empleo RB, aún a partir de datos incompletos. Las RB permiten el cálculo de la distribución de probabilidad condicional de la precipitación en función de diferentes localidades tanto cercanas como distantes. Estas observaciones confirman que las RB pueden ser usadas para identificar relaciones previamente indeterminadas entre variables climáticas en localidades distantes y cuantificarlas mediante inferencia con el objetivo de

realizar predicciones. Se sugiere la realización de estudios posteriores que impliquen otras variables climáticas como dirección de los vientos dominantes, presión atmosférica y temperatura del aire, que puedan ampliar la comprensión de los fenómenos meteorológicos y así realizar mejores predicciones.

Referencias

1. Cano R, Sordo C, Gutierrez J.M.: Applications of Bayesian Networks in Meteorology. In Gámez et al. (eds) *Advances in Bayesian Networks*, Springer, pp. 309-327 (2004)
2. Cofiño AS, Cano R, Sordo C, Gutierrez J.M.: Bayesian Networks for Probabilistic Weather Prediction: Proceedings of the 15th European Conference on Artificial Intelligence, pp. 695-700 (2002)
3. Correa M, Bielza C, Paimes-Teixeira J, Alique J.R.: Comparison of Bayesian networks and artificial neural networks for quality detection in a machining process. *Expert Syst. Appl.*, 36(3): 7270-7279 (2009)
4. Easterling D.R., Peterson T.C.: A new method for detecting undocumented discontinuities in climatological time series. *Int. J. Climatol.*, 15(4): 369-377 (1995)
5. Gámez J.A., Mateo J.L., Puerta J.M.: Learning Bayesian networks by hill climbing: efficient methods based on progressive restriction of the neighborhood. *Data Min. Knowl. Discov.*, 22: 106-148 (2011)
6. García, E (1988). Modifications to Köppen classification system climate to suit the conditions of the Mexican Republic. *Offset Larios S.A. Mexico*, pp. 201-217.
7. Garrote, L, Molina, M, Mediero, L.: Probabilistic Forecasts Using Bayesian Networks Calibrated with Deterministic Rainfall-Runoff Models. In Vasiliev et al. (eds.), *Extreme Hydrological Events: New Concepts for Security*, Springer, pp. 173-183 (2007)
8. Guoliang, L.: Knowledge Discovery with Bayesian Networks. PhD dissertation, National University of Singapore, Singapore (2009)
9. Guttman, N.B.: Statistical descriptors of climate. *Bull. Am. Meteorol. Soc.*, 70: 602-607 (1989)
10. HongRui, W., LeTian, Y., XinYi, X., QiLei, F., Yan, J., Qiong, L., Qi, T.: Bayesian networks precipitation model based on hidden Markov analysis and its application. *Sci China Tech. Sci.*, 53(2): 539-547 (2010)
11. Hruschka, E., Ebecken N.F.F.: Bayesian networks for imputation in classification Problems. *J. Intell. Inform. Syst.*, 29: 231-252 (2007)
12. INEGI 2011 www.inegi.org.mx
13. Kazemnejad, A., Zayeri, F., Hamzah, N.A., Gharaaghaji, R., Salehi, M.: A Bayesian analysis of bivariate ordered categorical responses using a latent variable regression model: Application to diabetic retinopathy data. *Sci. Res. Essays*, 5(11): 1264-1273 (2010)
14. Landsberg, H.E.: Weather 'normals' and normal weather. *Wkly. Weather Crop Bull.*, 42: 7-8 (1955)
15. Lima, C.H.R., Lall, U.: Spatial scaling in a changing climate: A hierarchical bayesian model for non-stationary multi-site annual maximum and monthly streamflow. *J. Hydrol.*, 383(3): 307-318 (2010)
16. Mediero, O.L.: Probabilistic forecast flood flows Through Bayesian Networks Applied to a Distributed Hydrological Model. PhD dissertation, Polytechnic University of Madrid, Madrid, Spain (2007)
17. Naveed, N., Choi, M.T.S., Jaffar, A.: Malignancy and abnormality detection of mammograms using DWT features and ensembling of classifiers. *Int. J. Phy. Sci.*, 6(8): 2107-2116 (2011)
18. Pearl, J.: Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann, San Mateo CA, United States, pp. 64-70 (1988)

19. Reyes, P.: Bayesian networks for setting genetic algorithm parameters used in problems of geometric constraint satisfaction. *Intell. Artificial.*, 45: 5-8 (2010)
20. Subramaniam, T., Jalab, H.A., Taqa, A.Y.: Overview of textual antispam filtering techniques. *Int. J. Phys. Sci.*, 5(12): 1869-1882 (2010)
21. Tae-wong, K., Hosung, A., Gunhui, C.H., Chulsang, Y.: Stochastic multi-site generation of daily rainfall occurrence in south Florida, *Stoch. Environ. Res. Risk Assess.*, 22: 705-717 (2008)

Modelado del problema de reparto utilizando un enfoque de planificación basado en Answer Set Programming

Adriana Huitzil Tello¹, Claudia Zepeda Cortés¹ y Mauricio Osorio Galindo²

¹ Benemérita Universidad Autónoma de Puebla,
Puebla, México
adryhut@gmail.com, czepedac@gmail.com

² Universidad de las Américas – Puebla,
Puebal, México
osoriomauri@gmail.com

Resumen La Planificación es una manera en la que un agente puede tomar ventaja del conocimiento que posee, y la habilidad de razonar acerca de las acciones y sus efectos en el ambiente. Answer Set Programming es un lenguaje de programación lógico y declarativo en el área de representación del conocimiento enfocado en la lógica. El propósito de este artículo consiste en usar la Planificación y Answer Set Programming para modelar y resolver el problema de reparto. Consideramos el caso donde un robot se encuentra en una oficina y tiene a su alcance ciertos paquetes, que a través de una secuencia de acciones, debe entregar a los lugares correspondientes de acuerdo a un horario de solicitud.

Palabras clave: Planificación, programación lógica, *Answer Set Programming*.

1. Introducción

La Programación Lógica o LP ofrece una manera más natural de representar conocimiento declarativo utilizando el lenguaje de la lógica matemática [1], construyendo una base de conocimientos a través de hechos y reglas entre diversos objetos que pueden ser capturados en un programa lógico, y con la ayuda de herramientas computacionales poder contestar preguntas sobre los objetos descritos y encontrar soluciones a problemas particulares.

La finalidad de un programa lógico es buscar todos los valores que hacen verdadero a este programa y como resultado de las combinaciones de dichos valores obtener un modelo estable [2]. Para que un programa lógico tenga un significado y pueda ser interpretado por programas computacionales se le debe asignar una semántica, la cual nos permite determinar el tipo de conclusiones que se pueden establecer a partir de un conjunto de reglas [1]. Debido a esto surgen lenguajes que permiten escribir programas lógicos que soportan dichas expresiones tales como el uso de Answer Set Programming. Answer Set Programming (ASP, Programación Lógica Estable o A-Prolog) es un área de representación del conocimiento enfocado en la lógica, basado en lenguajes para el modelado de problemas de cálculo en términos de restricciones

[3, 4]. La técnica de ASP se basa en la posibilidad de representar ciertas colecciones de conjuntos como colecciones de modelos estables [5]. El éxito de la semántica de los modelos estables, y posteriormente de los Answer Sets, se debió en gran medida a su capacidad para resolver problemas. Gracias a la existencia de implementaciones eficientes para calcular modelos estables tales como Smodels¹, DLV², Clasp³, enfocados a resolver problemas en base a la semántica estable, la gama de problemas que podían enfrentarse con este nuevo paradigma creció rápidamente para incluir problemas combinatorios, establecer y resolver problemas de planeación, modelar el comportamiento de agentes lógicos y aplicaciones de inteligencia artificial en general [1].

Un problema de Planificación en el que se puede hacer uso de Answer Set Programming es el problema de Secuenciación Mínima con Tiempos de Realización, que se define de la siguiente manera [6]: Sea T un conjunto de tareas, para cada tarea $t \in T$, una liberación de tiempo $r(t) \in \mathbb{Z}^+$, una longitud $l(t) \in \mathbb{Z}^+$ y un peso $w(t) \in \mathbb{Z}^+$. Un generador de tareas para T que obedece a un tiempo de realización, es decir, una función $f: T \rightarrow \mathbb{N}$ dado que, para todo $u \geq 0$, si $S(u)$ es el conjunto de tareas t para cada $f(t) \leq u < f(t) + l(t)$, entonces $|S(u)| = 1$ y para cada tarea t , $f(t) \geq r(t)$. Como se observa, lo que se establece es un conjunto de tareas, donde a cada tarea se le asigna una liberación de tiempo, además de establecer una longitud y un peso para cada una de ellas.

En este artículo presentamos la solución del problema de Secuenciación Mínima para el caso particular de reparto de oficios basado en un enfoque de Planificación lógica utilizando Answer Set Programming. El problema propuesto de Secuenciación Mínima con Tiempos de Realización que modelamos es el siguiente: *Dado un conjunto de objetos, estos deben ser entregados, a través de una secuencia de acciones, a lugares donde se solicite, en un tiempo indicado, llevando el control de los tiempos en que se realizan las acciones.*

La contribución de este trabajo es modelar el problema, generar una implementación y dar solución al problema de Secuenciación Mínima para un dominio particular (en este caso el reparto de oficios) utilizando un enfoque de planificación basado en Answer Set Programming.

En relación con Answer Set Programming y Planning, actualmente se han realizado diversos proyectos, uno de los más recientes es el modelado de espacio de datos autónomo usando Answer Sets, que está basado en la expresividad de los modelos estables y la acción del lenguaje k para expresar las funciones de gestión de un espacio de datos, el cual se basa en un modelo para especificar un espacio de datos autónomo que son expresados mediante Answer Sets [7]. Por otra parte también se ha aplicado Planeación en Answer Set como es el caso de [8], el cual permite simular y dar apoyo en la generación de nuevos planes de evacuación mediante el uso de Answer Set Programming para describir una zona de riesgo y permitir resolver consultas que no pueden ser resueltas por un Sistema de Información Geográfica (SIG), otro ejemplo de este tipo se muestra en [9], en donde se modelan problemas de evacuación para situaciones de riesgo. Un caso similar al que tratamos en este artículo es

¹ <http://www.tcs.hut.fi/Software/smodels/>

² <http://www.dbai.tuwien.ac.at/proj/dlv/>

³ <http://www.cs.uni-potsdam.de/clasp/>

el Modelado y planeación con Answer Set Programming [10], también se hace uso de preferencias en Answer Set como es el caso de [11] y [12].

En la sección 2 se describen conceptos que se consideran importantes para resolver el problema de reparto de oficios así como algunos trabajos que se relacionan con el trabajo a desarrollar en este artículo. En la sección 3 establecemos una descripción a detalle del problema de reparto de oficios. En la sección 4 se muestra la codificación del problema en el lenguaje A. En la sección 5 se realiza la codificación del problema utilizando la herramienta Smodels. En la sección 6 se indican algunas diferencias de codificación en Clasp con respecto a Smodels, y por último en la sección 7 presentamos las conclusiones.

2. Marco teórico

En esta sección presentamos definiciones básicas para el desarrollo del problema de Secuenciación Mínima.

2.1. Programación lógica

La Programación Lógica trabaja de una forma descriptiva, es decir, estableciendo relaciones entre entidades (objetos del mundo), sin indicar cómo se debe hacer, sino más bien qué hacer, ya que es el programador quien suministra la parte lógica y el intérprete la parte del control [13]. Un programa lógico está formado por un conjunto de cláusulas, que son sentencias descriptivas que puede ser simples o compuestas por otras cláusulas, además de contener operadores binarios, por ejemplo: and (\wedge), or (\vee), implicaciones (\leftarrow), negaciones (not, \neg), etc. [2].

2.2. Lenguaje A

El *lenguaje A* es un elemento muy importante dentro de la Programación Lógica, ya que es el pseudocódigo de un programa para Planning. El alfabeto del lenguaje A consiste de 2 conjuntos disjuntos de símbolos no vacíos que son F y A [3], los cuales corresponden al conjunto de Fluents y al conjunto de Acciones respectivamente.

Los *fluents* forman parte de la descripción del estado del mundo y representan las propiedades de los objetos en el mundo. Un *fluent literal* es un fluent, el cual puede estar precedido por la negación \neg . Las acciones, representan las operaciones aplicadas a los fluents que permiten realizar cambios de estados. El lenguaje de acción A está representado a través de tres sub-lenguajes que son: *lenguaje de descripción del dominio*, *lenguaje de observación* y *lenguaje de consulta* [3, 14].

Lenguaje de descripción del dominio. Indica lo que ocurre entre estados al momento de ejecutar las acciones. La descripción del dominio D consiste en proposiciones de efecto que son de la siguiente forma: $a \text{ causes } f \text{ if } p_1, \dots, p_m \neg q_1, \dots, \neg q_r$ donde a es una acción, f y $p_1, \dots, p_m, q_1, \dots, q_r$ son fluents, esto significa que si los fluents literales $p_1, \dots, p_m \neg q_1, \dots, \neg q_r$ se cumplen en el estado correspondiente a una situación s , entonces en el estado correspondiente a la situación alcanzada por ejecutar a en s , el fluent literal f debe cumplirse. También se puede incluir ejecutabilidad de condiciones, de la forma: **ejecutable** $a \text{ if } p_1, \dots, p_m \neg q_1, \dots, \neg q_r$ donde a es una acción y, $p_1, \dots, p_m, q_1, \dots, q_r$

son fluents, esto significa que si los fluents literales $p_1, \dots, p_n, \neg q_1, \dots, \neg q_r$ se cumplen en el estado σ de una situación s , entonces la acción a es ejecutada en s .

Lenguaje de observación: Es un conjunto de observaciones O , formado por valores de proposiciones de la siguiente forma [15]: f **after** a_1, \dots, a_m en donde f es un fluent y a_1, \dots, a_m son las acciones. Esta proposición significa que si las acciones a_1, \dots, a_m se ejecutan en el estado inicial entonces en el estado correspondiente a la situación $[a_1, \dots, a_m]$ el fluent f se cumple. En otro caso, cuando la secuencia de acciones a_1, \dots, a_m es una secuencia vacía, únicamente se indica el fluent marcándolo como estado inicial: **initially** f .

Lenguaje de consulta. En un dominio consistente de descripción D , en la presencia de un conjunto de observaciones O implica una consulta Q de la forma: f **after** a_1, \dots, a_m si para todos los estados iniciales σ_0 que corresponde a (D, O) , el fluent literal f se cumple en el estado $[a_m, \dots, a_1] \sigma_0$. Lo cual denotamos como $D \models_O Q$.

2.3. Resolvedores de ASP: Smodels y Clasp

Actualmente, existen varios resolvedores para Answer Set, por cuestiones de espacio solo mencionaremos dos de ellos: Smodels y Potassco⁴ en particular la herramienta Clasp, los cuales se utilizarán para obtener los Answer Sets del problema de Secuenciación Mínima. Smodels es una eficiente implementación de la semántica de modelos estables para programas lógicos. Clasp es un sistema solucionador de Answer Set para extender programas lógicos normales.

3. Descripción del problema de reparto de oficios

En esta sección se presenta la descripción para el caso particular del problema de Secuenciación Mínima, el cual consiste en repartir oficios. El problema está enfocado en un robot que tiene como principal tarea llevar oficios a los empleados de las diferentes oficinas. Los elementos que componen el mundo para el problema de reparto de oficios son los siguientes: un robot que es el encargado de entregar los oficios, las oficinas donde se realizarán las entregas y la recepción donde el robot tomará los oficios. Inicialmente, el robot está en la recepción y los oficios se encuentran sobre una mesa los cuales están organizados en carpetas, cada carpeta tiene el nombre del empleado al cual deberán dirigirse los oficios, por lo tanto, si el robot se encuentra en la recepción solo debe tomar los oficios correspondientes, teniendo claro que eso representa la carpeta con el nombre del empleado al que entregará los oficios. Posteriormente el robot deberá entregar los oficios realizando una secuencia de acciones definidas y en base a los horarios que el empleado ha indicado para recibir sus oficios, tomando en cuenta los tiempos que ocupa para trasladarse del estado inicial a cada oficina y viceversa, los tiempos de ida y vuelta deben ser los mismos. El robot solo puede llevar una carpeta de oficios y realizar una entrega en un tiempo determinado.

En este caso se modela la entrega de oficios en una oficina, sin embargo, este modelo puede amoldarse a la entrega de diversos objetos como: herramienta, instrumental, paquetes, sobres, café, etc., y además puede crecer en base al número de oficinas.

⁴ <http://potassco.sourceforge.net/>

Las pruebas que realizamos para la entrega de oficios fueron de una a cinco oficinas, las cuales se muestran en el siguiente sitio: <https://sites.google.com/site/codigoentregadeoficios/>

Primero se muestra la codificación en Smodels que fue la primera parte con la que trabajamos realizando varias pruebas para obtener una versión final, y posteriormente la codificación en Clasp de estas mismas pruebas, en ambos casos se obtuvieron los Answer Sets de cada prueba para comparar los modelos obtenidos y analizar las variantes de cada una.

4. Código del problema de reparto de oficios en el lenguaje A

En esta sección haremos uso del lenguaje A para modelar el problema de Secuenciación Mínima que hemos planteado anteriormente, el cual consiste en entregar oficios en una oficina:

Fluentes.

```
tiempoSalida(T). % Representa el tiempo en que el robot sale de un lugar.
tiempoLlegada(T). % Representa el tiempo en que el robot llega a un lugar.
estaEnLugar(L). % Indica el lugar en que se encuentra el robot
robotTieneOficios. % Indica que el robot tiene los oficios que va a entregar.
entregoOficiosEnLugar(L). % El robot ha entregado los oficios en el lugar indicado.
```

Acciones.

```
move(L1,L2). % Representa que el robot se moverá del lugar L1 al lugar L2.
tomarOficios. % Indica que el robot debe tomar los oficios.
entregarOficios(L,T_entrega). % Entregar oficios en el tiempo indicado.
```

Efectos y ejecutabilidad de condiciones.

```
move(L1,L2)
  causes estaEnLugar(L2)
  if holds(estaEnLugar(L1),T), T < length, time(T),
      lugar(L1), lugar(L2), neq(L1,L2).
```

Esta regla significa que el robot se moverá de L1 a L2 causando que esté en L2 si se cumple que el robot se encuentra en L1 en el tiempo T, y L1 y L2 son lugares diferentes. Para la ejecución de la acción tomar oficios será de la siguiente manera:

```
tomarOficios
  causes robotTieneOficios
  if holds(estaEnLugar(recepcion),T),
      holds(neg(robotTieneOficios),T), T < length,
```

* El símbolo % es usado para indicar un comentario dentro del código

```
time(T).
```

Esto indica que si el robot ejecuta la acción de tomar oficios causará que el robot tenga los oficios, si se cumple que se encuentra en la recepción y aún no tiene los oficios a entregar. La última ejecución es para la acción entregar oficios:

```
entregarOficios(L,T_entrega)
causes neg(robotTieneOficios), entregoOficiosEnLugar(L)
if T < length, time(T), holds(estaEnLugar(L),T),
    holds(robotTieneOficios,T),
    holds(neg(entregoOficiosEnLugar(L)),T),
    holds(tiempo_llegada(T_llegada),T),
    horario_oficios(L,T0,T1), T0<= T_llegada,
    T_llegada <= T1, assign(T_entrega,T_llegada),
    lugar(L), timeh(T_entrega), timeh(T_llegada),
    timeh(T0), timeh(T1).
```

de esta manera se indica que si el robot ejecuta la acción entregar oficios causará que el robot ya no tenga los oficios y que los ha entregado en la oficina, si se cumple que el robot está en el lugar donde realizará la entrega, tiene los oficios a entregar, aún no ha entregado los oficios en ese lugar, se conoce el tiempo en que el robot llegó al lugar para que a través del horario que el empleado indicó se realice una comparación entre el tiempo de llegada y los tiempos establecidos, y si se cumple esta comparación se asigne el tiempo de llegada al tiempo de entrega con el objetivo de no perder el conteo del tiempo.

Estado inicial.

El estado inicial muestra los elementos del mundo antes de sufrir una modificación:

```
initially estaEnLugar(recepcion). %El robot está en la recepción.
initially neg(robotTieneOficios). %El robot aún no tiene los oficios.
initially tiempoSalida(1). %El robot inicia en el tiempo 1.
initially neg(entregoOficiosEnLugar(oA)). %Aún no se han entregado los oficios en el lugar indicado.
```

Estado final.

El estado final representa la meta del plan indicado de esta forma:

```
finally entregoOficioEnLugar(oA). %Entrega de oficios en la oficina.
finally estaEnLugar(oA). %El robot está en la oficina oA.
```

5. Código del problema de reparto de oficios en Smodels

Siguiendo con lo planteado anteriormente, se muestra el código correspondiente a la entrega de oficios en una oficina utilizando Smodels.

Declaración de constantes y tiempos.

```
const length = 4. % Declaración de la constante length.
time(1..length). % Indica el tiempo máximo para llegar a la meta.
const reloj = 4. % Declaración de la constante reloj.
timeh(0..reloj). % Indica la longitud de los pasos para realizar las acciones.
```

Declaración de hechos.

```
lugar(recepcion). % Declaración del lugar recepción.
lugar(oA). % Declaración del lugar oficina de Ana.
horario_oficios(oA,2,3). %Horario de preferencia para recibir los oficios.
tiempo_viaje(recepcion,oA,1). %Tiempo para trasladarse a la oficina.
tiempo_viaje(oA,recepcion,1). %Tiempo para trasladarse a la recepción.
```

Fluentes.

```
fluent(tiempo_salida(T)) :- timeh(T). %Tiempo de salida del lugar.
fluent(tiempo_llegada(T)) :- timeh(T). %Tiempo de llega al lugar.
fluent(esta_en_lugar(L)):- lugar(L). %Lugar en donde está el robot.
fluent(robot_tiene_oficios). %El robot tiene los oficios a entregar.
fluent(entrego_oficios_en_lugar(L)) :- lugar(L). %El robot ha entregado los oficios en el lugar indicado.
```

Acciones.

```
action(move(L1,L2)):- lugar(L1), lugar(L2). %Moverse del lugar L1 al lugar L2.
action(tomar_oficios). %Tomar los oficios.
action(entregar_oficios(L,T_entrega)):- timeh(T_entrega),
lugar(L). %Hacer entrega de los oficios en el tiempo de entrega establecido.
```

Estado inicial.

```
initially(esta_en_lugar(recepcion)). %El robot está en la recepción.
initially(neg(robot_tiene_oficios)). %El robot no tiene los oficios.
initially(tiempo_salida(1)). %Indica que el robot inicia en el tiempo 1.
initially(neg(entrego_oficios_en_lugar(oA))). %Aún no se han entregado los oficios en el lugar indicado.
```

Estado final.

El estado final representa la meta del plan indicado de esta manera:

```
finally(entrego_oficios_en_lugar(oA)). %Entrega de oficios en la oficina oA.
finally(esta_en_lugar(oA)). %El robot está en la oficina oA.
```

Declaración de las condiciones de ejecutabilidad.

```

executable(move(L1,L2),T) :- T < length, time(T),
    holds(esta_en_lugar(L1),T),
    lugar(L1), lugar(L2), neq(L1,L2).

```

Esta regla significa que el robot se moverá de L1 a L2 si se cumple que el robot se encuentra en L1 en el tiempo T y L1 y L2 son lugares diferentes. Para la ejecución de la acción tomar oficios será de la siguiente manera:

```

executable(tomar_oficioS,T) :- T < length, time(T),
    holds(esta_en_lugar(recepcion),T),
    holds(neg(robot_tiene_oficios),T).

```

esto indica que el robot ejecuta la acción de tomar oficios si se cumple que el robot se encuentra en la recepción y aún no tiene los oficios. La última ejecución es la acción de entregar oficios:

```

executable(entregar_oficios(L,T_entrega),T):- T < length,
    time(T), lugar(L), timeh(T_entrega),
    holds(esta_en_lugar(L),T),
    holds(robot_tiene_oficios,T),
    holds(neg(entrego_oficios_en_lugar(L)),T),
    holds(tiempo_llegada(T_llegada),T),
    horario_oficios(L,T0,T1), T0<= T_llegada,
    T_llegada <= T1, assign(T_entrega,T_llegada),
    timeh(T_llegada), timeh(T0), timeh(T1).

```

de esta manera, indicamos que el robot ejecuta la acción entregar oficios si se cumple que el robot está en el lugar donde realizará la entrega, tiene los oficios a entregar, y aún no ha entregado los oficios en ese lugar, se conoce el tiempo en que el robot llegó al lugar para que a través del horario que el empleado indicó se realice una comparación entre el tiempo de llegada y los tiempos establecidos, y si se cumple esta comparación se asigne el tiempo de llegada al tiempo de entrega con el objetivo de no perder el conteo del tiempo.

Reglas de causa.

```

causes_dd(esta_en_lugar(L2),move(L1,L2),
esta_en_lugar(L1)):- lugar(L1), lugar(L2).

```

Esta regla significa que si el robot está en el lugar L1 y se ejecuta la acción mover de L1 a L2 entonces causa que el robot esté en el lugar L2. El caso contrario de esta regla es:

```

causes_dd(neg(esta_en_lugar(L1)),move(L1,L2),
esta_en_lugar(L1)) :- lugar(L1), lugar(L2), neq(L1,L2).

```

que quiere decir, que si el robot está en el lugar L1 y se ejecuta la acción mover de L1 a L2 entonces causa que el robot ya no esté en ese el lugar L1. Existen otras re-

glas de causa de la misma forma que la anterior para la acción mover, en donde el efecto es obtener el tiempo de llegada y negar de igual forma el tiempo de llegada y salida. Esto se podrá observar en el sitio ya mencionado debido a que no se tiene más espacio para escribirlas.

Para la acción tomar oficios tenemos la siguiente regla de causa:

```
causes(tomar_oficios, robot_tiene_oficios).
```

esta regla indica que si el robot toma los oficios causará que el robot tenga los oficios a entregar. Para la acción de entregar oficios tenemos dos reglas de la misma forma que se hace para la acción tomar oficios, en donde negamos que el robot tiene los oficios y que ya entregó los oficios, esto podemos observarlo en el sitio mencionado.

6. Código del problema de reparto de oficios en Clasp

El código correspondiente de la aplicación en Clasp es muy parecido al de Smodels, de tal forma que por cuestiones de espacio, solo se mostrarán las diferencias que tiene Clasp con respecto a Smodels, las cuales son:

Declaración de constantes y tiempos.

```
#const length = 4. % Declaración de la constante length.
time(1..length). % Indica el tiempo máximo para llegar a la meta.

#const reloj = 4. % Declaración de la constante reloj.
timeh(0..reloj). % Indica la longitud de los pasos para realizar las acciones.
```

Declaración de hechos.

```
lugar(recepción;oA). %Declaración de 2 lugares recepción y oficina de Ana.
```

La declaración de fluentes, acciones, estados iniciales y finales y reglas de causa se hace de la misma manera que en Smodels.

Declaración de las condiciones de ejecutabilidad.

En este caso, se utilizarán reglas llamadas restricciones de cardinalidad, que indican las posibles combinaciones que se pueden hacer en un conjunto de valores. Éstas se definen de la siguiente forma:

```
1{executable(tomar_oficios,T) : T < length: time(T) }1.
executable(tomar_oficios,T) :-
    holds(esta_en_lugar(recepcion),T),
    holds(neg(robot_tiene_oficios),T).
```

esta regla de ejecución indica tomar oficios en un tiempo T, donde T sea menor a la constante length y T sea de tipo time, si el robot está en la recepción y no tiene los oficios. Para la ejecución de la acción entregar oficios se define de manera similar,

indicando entre llaves los tipos de las variables a utilizar para la ejecución. Para ejecutar la acción mover se tiene lo siguiente:

```
1{executable(move(L1,L2),T) : T < length: time(T) : lugar(L1):  
lugar(L2)} 1.
```

lo que indica, que el robot se moverá de L1 a L2 en el tiempo T, donde L1 y L2 son lugares diferentes.

Para el caso de 5 oficinas se podrá observar con mayor precisión el uso de las condiciones de ejecutabilidad. Las soluciones de ambas implementaciones se observan en el sitio descrito anteriormente donde se encuentran todos los códigos en ambas versiones junto con sus ejecuciones, las cuales muestran de una manera más clara los resultados obtenidos en cada implementación.

7. Conclusiones

Al ejecutar estos códigos en Smodels y Clasp, se observó que la entrega de oficios de una hasta cinco oficinas se realiza satisfactoriamente y para el tiempo de resolución de Answer Set, Clasp los obtiene más rápido. Por otra parte, la forma de codificar en Smodels y Clasp es muy parecida, pero Clasp permite realizar una reducción de código. En Smodels fue un tanto difícil codificar porque no se conocía a fondo la sintaxis de esta herramienta y en Clasp fue más fácil por la experiencia adquirida.

Agradecimiento. Este trabajo ha sido apoyado por el Proyecto de Ciencia Básica del Fondo Sectorial SEP-CONACyT con número de registro 101581.

Referencias

1. Juan Antonio Navarro: Lógica Aplicada a Answer Sets. Tesis de Licenciatura. Universidad de las Américas Puebla. Online <http://www.mpi-sws.org/~jnavarro/papers/uthesis.pdf> (última fecha de verificación mayo 2012), páginas 61 (2003)
2. Guillermo Didier Bravo: Desarrollo y modelado de algoritmos para la genotipificación de secuencias: el caso de los trasplantes, alelos HLA. Tesis de Licenciatura. Universidad de las Américas Puebla. Online http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/bravo_a_gd/capitulo3.pdf (última fecha de verificación mayo 2012), páginas 77 (2005)
3. Chitta Baral: Knowledge Representation, Reasoning and Declarative Problem Solving with Answer Sets. Cambridge University Press, Cambridge (2003)
4. Iikka Niemelä: Logic programming with stable model semantics as a constraint programming paradigm. *Annals of Mathematics and Artificial Intelligence* 25(3-4) 241–273 (1999)
5. <http://www.ia.urjc.es/cms/sites/default/files/userfiles/file/ia3/teoria/Intro-ASP.pdf>
6. G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti Spaccamela and M. Protasi: Complexity and Approximation. *Combinatorial Optimization Problems and Their Approximability Properties*. Springer, páginas 436 (2003)
7. Gabriela Montiel Moreno, Zechinelli Martini José Luis and Genoveva Vargas Solar: Modelling autonomic dataspace using answer sets. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, Volumen 14. Número 48. Online <http://redalyc.uaemex.mx/src/inicio/ArtPdfRed.jsp?iCve=92513175002> (2010)

8. Claudia Zepeda and David Sol: Evacuation Planning using Answer Set Programming: An initial approach. http://www.engineeringletters.com/issues_v15/issue_2/EL_15_2_10.pdf (última fecha de verificación mayo 2012) (2012)
9. José Miguel Leyva Cruz: Representación de Planes de Evacuación usando Answer Set Programming. Tesis de Ingeniería. Universidad Tecnológica de la Mixteca. Online http://jupiter.utm.mx/~tesis_dig/10107.pdf (última fecha de verificación mayo 2012), páginas 155 (2007)
10. Matilde Hernández Salas: Modelado y planeación con ASP (Answer Set Programming). Tesis de Maestría. Universidad de las Américas Puebla, 2004. Online http://catarina.udlap.mx/u_dl_a/tales/documentos/msp/hernandez_s_m/portada.html (última fecha de verificación mayo 2012) (2004)
11. Diego Rodríguez Villanueva: Utilización de los enfoques para Preferencias en Answer Set Programming para definir familias de problemas con preferencias. Tesis de Ingeniería. Universidad Tecnológica de la Mixteca. Online http://jupiter.utm.mx/~tesis_dig/10300.pdf (última fecha de verificación mayo 2012) (2007)
12. Alexander Gelbukh and Sergio Suárez Guerra: Special Issue: Advances in Computer Science and Engineering, 23:3-13. Online <http://www.micai.org/racs/Vol23.pdf> (2006)
13. Ravi Sethi: Programming Languages: Concepts & Constructs. 2nd Ed. Addison-Wesley (1997)
14. Michael Gelfond and Vladimir Lifschitz: Action languages. Electron. Trans. Artif. Intell. 2:193-210 (1998)
15. V. Subrahmanian and C. Zaniolo: Relating stable models and AI planning domains. In: Proceedings of the 12th International Conference on Logic Programming, L. Sterling, Pages 233-247. Tokyo, Japan, MIT Press (1995)

Implementación de la estrategia de juego “Tic-Tac-Toe” para la interacción con un brazo robótico

Dora-Luz Almanza-Ojeda¹, Francisco Luna-Rodríguez¹,
David Isaac Hernández Santos¹, Esteban Pérez Flores¹ y
Mario-Alberto Ibarra-Manzano²

¹ Universidad Politécnica de Guanajuato,
Av, Universidad Norte S/N Comunidad Juan Alonso
Cortazar, Guanajuato, México, 38483

luzdora@ieee.org
{luis.luna.rdz, disaachs, steve.epf}@gmail.com
<http://www.upgto.edu.mx>

² Universidad de Guanajuato,
Carretera Salamanca-Valle de Santiago Km.3.5+1.8Km,
Salamanca, Guanajuato, 36885, México

ibarram@ugto.mx
<http://www.ugto.mx>

Resumen En este artículo se presenta un sistema para la interacción hombre-máquina, el cual consiste de un algoritmo que permite a un robot manipulador jugar el clásico juego de “Tic-Tac-Toe” con un humano. El estado actual del tablero de juego es proporcionado al robot mediante una cámara color. El análisis de imágenes adquiridas proveen al robot la información sobre el desarrollo del juego. Se describe la sincronización entre el módulo de control del robot, el sistema de visión y el algoritmo desde una estación de trabajo. El análisis de desempeño del algoritmo y del sistema integrado en funcionamiento es presentado. Al final del documento se presentan las conclusiones y perspectivas.

Palabras clave: Robótica, algoritmo Tic-Tac-Toe, visión por computadora, interacción hombre-máquina, sistemas inteligentes.

1. Introducción

La automatización de tareas en robots manipuladores es un área muy explotada a nivel industrial y de investigación. Dichas tareas requieren diversos conocimientos de ingeniería, como son programación, control, procesamiento de señales, procesamiento de imágenes, entre otras. En el interés de acercar el desarrollo de proyectos multidisciplinarios en la formación de nuevas generaciones de ingenieros, comúnmente, se involucran plataformas robóticas y la programación modular con otros dispositivos. Aunque el desarrollo de inteligencia artificial para

robots ha sido bastante aplicada recientemente con fines educativos, nuevas estrategias de solución e implementación de sistemas inteligentes, permiten que siga siendo un recurso importante para la enseñanza.

Proponer estrategias de interacción entre sistemas robóticos y personas, en este caso, para realizar un juego de “Tic-Tac-Toe”, resulta en un proyecto motivante que muestra de forma sencilla, pero completa, las diferentes etapas involucradas para alcanzar el objetivo. A pesar de que, este proyecto ha sido ya presentado para diferentes tipos de robots manipuladores, aún continúa el interés por presentar estrategias diferentes y mejoradas de estos sistemas, gracias a la mejora continua del desempeño en los equipos de cómputo. Este hecho permite proponer algoritmos de decisión basados en el análisis de grandes cantidades de información, para el diseño de sistemas inteligentes más robustos.

En este trabajo se describe el desarrollo de un sistema robótico para jugar “Tic-Tac-Toe” con una persona, utilizando información visual. Este sistema involucra, desde la programación de un algoritmo de solución al juego, para la toma de decisiones en tiempo real, hasta la integración de diversos módulos y su sincronización para el funcionamiento satisfactorio del sistema.

2. Trabajos relacionados

Un área importante en el desarrollo de la inteligencia artificial son los algoritmos genéticos. En [2] y [5], los autores proponen una estrategia de juego “Tic-Tac-Toe” basada en algoritmos genéticos, con el fin de evolucionar la población inicial hacia la optimización deseada, en este caso, una estrategia donde el algoritmo no pierda. En este trabajo, los autores aprovechan la simetría del tablero para reducir de manera considerable el número de estados posibles en el juego. Sin embargo, a pesar de dicha reducción alcanzada, el uso de estas estrategias no es posible en sistemas que requieren respuesta en tiempo real.

Trabajos presentados en [1] y [7], muestran también la interacción del robot con una persona para realizar una partida de juegos de mesa. Estos proyectos son descritos como motivantes y muy apropiados para el desarrollo académico de estudiantes en ingeniería. Con el mismo fin académico, en [3] y [6] los autores se enfocan en el manejo del robot a distancia a través de una plataforma web. Se sabe que tales plataformas robóticas educativas, no importando que tan sencilla y adaptable sean en su uso, son principalmente cerradas a la programación abierta hecha por el usuario, limitando el control del robot de forma externa. Recientemente, en [4], los autores proponen una librería de tipo código abierto, mejor conocida como “Open Source”, con la cual es posible desarrollar el control de cada articulación del robot y con ello implementar trayectorias especificadas por el usuario.

Por otra parte, aunque varios sistemas robóticos de interacción proponen el análisis de imágenes para conocer la información sobre el tablero de juego, la descripción detallada de las técnicas de procesamiento de la imagen es sólo muy ligeramente introducida. Parte de este trabajo consiste en presentar una

propuesta sobre el análisis de las imágenes y la entrada de datos al sistema robótico que eviten el problema de cambios de iluminación y calibración [1].

Generalmente, la interacción de manipuladores con personas ha sido presentada para diferentes plataformas. Sin embargo, en nuestra experiencia las soluciones propuestas anteriormente, pueden ser todavía mejoradas en el aspecto de desempeño y el modo de procesamiento de la información de entrada, es por ello que decidimos exponer nuestras propuestas al respecto.

Los principales aspectos discutidos en este artículo son:

- Análisis de imágenes
- Algoritmo de juego
- Comunicación entre módulos
- Desempeño

3. Diagrama global del sistema

Nuestro sistema de juego de “Tic-Tac-Toe” en interacción con un brazo robótico, propone el uso de un tablero dividido en 9 celdas formadas por 3 renglones y 3 columnas. En lugar del clásico símbolo del jugador “X” ó “O”, se utilizan fichas de 2 colores diferentes, cada color distingue al jugador. Además, el desarrollo de una partida de juego, requiere una computadora, una cámara, un tablero y las fichas de juego y un robot y su controlador. La relación de estos queda descrita en el diagrama esquemático de la figura 1.

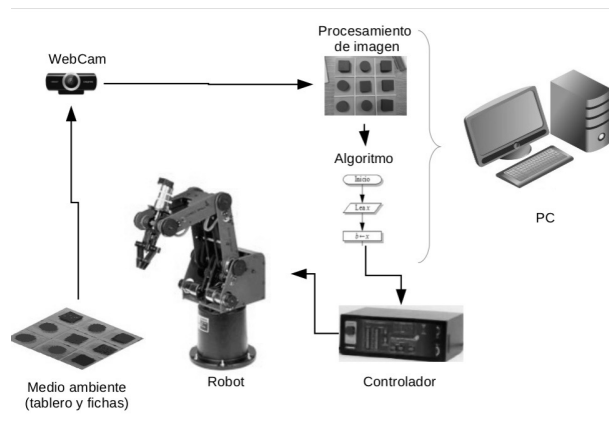


Figura 1. Diagrama global del sistema

La imagen capturada desde la cámara web proporciona el estado del tablero. El procesamiento de la imagen se lleva a cabo en la PC mediante algoritmos de filtrado de imágenes para la detección del color, lo cual va constituir la entrada de datos al algoritmo. Una vez que la estrategia de la siguiente jugada a realizar

con el SCORBOT es definida por el algoritmo, entonces se envían las órdenes correspondientes al controlador del robot. Cabe mencionar que el tablero fue diseñado de acuerdo con el espacio de trabajo del robot, se realizaron fichas de diferentes colores con los cuales se realizaron las pruebas del sistema completo. Finalmente, se utilizó MatLab para desarrollar la programación del algoritmo y el procesamiento de la imagen. Los movimientos del robot para colocar las fichas sobre el tablero, fueron codificadas usando el lenguaje propio del SCORBOT. La sección siguiente presenta una descripción general de cada módulo del sistema.

4. Descripción modular del sistema

La integración del sistema requiere de los siguientes componentes: el módulo de visión, el algoritmo de juego, el módulo del controlador y el robot. El algoritmo de juego puede definirse como el más importante de todos, ya que permite que nuestro robot se comporte como un sistema capaz de tomar una decisión de manera autónoma, dependiendo del escenario de juego. La estrategia de juego se basa en realizar jugadas de ataque sobre el centro y las esquinas, debido a que éstas permiten una más alta probabilidad de triunfo sobre el resto de las casillas en el tablero. Así, el centro permite realizar 4 de las 8 posibles jugadas de gane, siendo ésta la más alta probabilidad dada por una casilla. Después, las casillas de las esquinas permiten realizar jugadas de triunfo en sentidos ortogonales o sobre la diagonal. Finalmente, el resto de las casillas permite solo triunfos en sentidos ortogonales. A continuación, con el fin de describir los diferentes módulos del sistema y la dinámica de juego, nos referiremos a los jugadores como el usuario (jugador “O”) y el SCORBOT (jugador “X”).

4.1. Detección de fichas en el tablero mediante análisis de imágenes

El propósito de un sistema de visión consiste en cuantificar, de manera automática y sin intervención humana, las características físicas de un objeto y extraer información del entorno para controlar un proceso. Éste queda conformado por una o más fuentes de iluminación, una cámara y su óptica y un equipo de cómputo para realizar el procesamiento de la imagen. En este proyecto aprovechamos las ventajas de usar un sistema de visión para adquirir información del escenario de juego. La figura 2(a) muestra una imagen del tablero vacío adquirida desde la cámara. La detección inicial de los bordes permite identificar los límites de las nueve regiones de interés. Tales regiones son enumeradas como se muestra en la figura 2(b) y son utilizadas en el algoritmo de decisión.

La figura 3 muestra un ejemplo del tablero en algún turno de la partida. De la imagen de entrada inicial, figura 3(a), obtuvimos los valores RGB de las fichas en el tablero. Utilizando la distancia euclidiana de cada pixel con respecto a un modelo establecido del color, se obtiene una imagen de probabilidad, figura 3(b). En esta figura los tonos oscuros representan la probabilidad más alta en similitud con el modelo de color establecido. Los tonos claros en la imagen de probabilidad, indican la presencia del color opuesto al modelo de referencia en

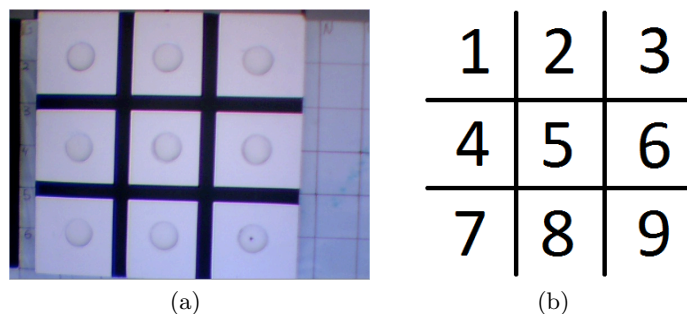


Figura 2. a) Imagen del tablero de prueba vacío, b) Numeración de casillas del tablero.

el espacio RGB. Finalmente, aplicando una umbralización a la imagen de probabilidad, se obtiene una imagen binaria, en donde, los píxeles blancos indican la posición de las fichas con el color buscado, tal como se muestra en la figura 3(c).

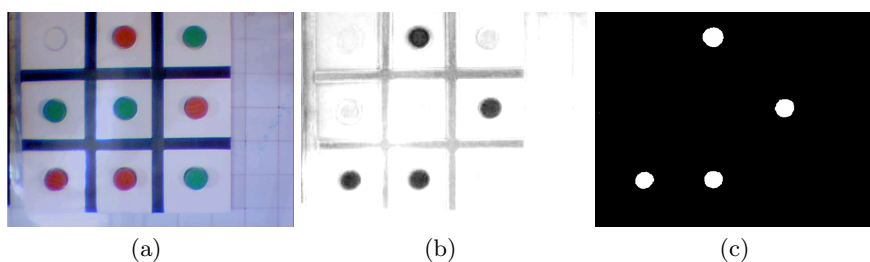


Figura 3. Etapas del procesamiento de la imagen. a) Imagen RGB inicial, b) Imagen de probabilidad de detección del color (rojo en este caso), c) Máscara binaria final

De la imagen binaria son detectadas las fichas en cada una de las nueve regiones de interés. En base a un conteo del número de “1’s” (color blanco) en cada una de las regiones, es posible identificar la presencia de la ficha del modelo de color de referencia. Este proceso es realizado para cada color con el objetivo de detectar, el color de la ficha y en cuál región se encuentra, de esta manera es obtenido el estado actual del tablero.

Con el fin de optimizar el tiempo de procesamiento deben tomarse en cuenta los siguientes puntos:

1. El campo de visión de la cámara debe cubrir solamente el área correspondiente al tablero para optimizar la información en la imagen.
2. El análisis de detección de las fichas debe hacerse solamente en las regiones de interés para reducir el tiempo de procesamiento.
3. El conteo de “1’s” para la detección de fichas debe de hacerse de forma horizontal, con el fin de reducir el número de puntos a analizar.

4.2. Algoritmo de “Tic-Tac-Toe” o tres en raya

El algoritmo desarrollado para este sistema es una modificación del propuesto en [8]. Éste plantea una estrategia sencilla pero eficiente, al momento de verificar posibles triunfos tanto del robot como del usuario, evitando una situación de derrota y buscando el posible triunfo del sistema. Sin embargo, realizando un análisis más a detalle de ese algoritmo, se encontró su principal debilidad cuando el usuario comienza jugando en las casillas pares como la 2, 4, 6 u 8. Como ya se mencionó anteriormente, dichas casillas no son de gran peso cuando se juega al “Tic-Tac-Toe”, ya que las esquinas y el centro proporcionan mayores posibilidades de triunfo. Esto constituía un lado muy vulnerable del método propuesto en [8], por lo cual debimos agregar ciertas modificaciones, resultando el algoritmo presentado en la tabla 1. Ciertos valores son asignados dependiendo del estado de la casilla, esto es: el número 2 indica vacío, el 3 indica que la casilla contiene una ficha de color rojo y el 5 indica que contiene una ficha de color verde. Por lo tanto, la metodología definida en cada turno dependerá del estado del tablero, para lo cual, el color rojo siempre es asignado al robot.

Tabla 1. Estrategía de juego “Tic-Tac-Toe”. Aquí se considera que “X” es el SCOR-BOT (estado 3 en el tablero) y “O” es el usuario (estado 5 en el tablero).

Turno	Estrategía
1	$mover(n)$, donde n es cualquiera de las esquinas
2	Si casilla[5] está vacía, $mover(5)$, si no $mover(n)$ donde n es cualquiera de las esquinas que este vacía
3	$mover(n)$, donde n es cualquiera de las esquinas vacías
4	Verificar si los siguientes pares de casillas están ocupadas simultáneamente: (2,6), (6,8), (8,4), (4,2), entonces mover a casilla 3, 9, 7 ó 1 de forma respectiva. Si en este turno se tiene como estados [3,5,5] en 3 casillas consecutivas o de las diagonales, esto es, donde la multiplicación sea 75, entonces el robot mueve la pieza a cualquiera de las esquinas que se encuentre desocupada. Si $posible_triunfo(X)$ no regresa un valor de 0, entonces $mover(posible_triunfo(X))$.
5	Si $posible_triunfo(X)$ no regresa un valor de 0, entonces $mover(posible_triunfo(X))$ (victoria de “X”), sí no, si $posible_triunfo(O)$ no regresa un valor de 0, entonces $mover(posible_triunfo(O))$ (se evita victoria de “O”), de otra manera $mover(n)$ donde n es cualquier casilla vacía
6	Se repite turno 5
7	Se repite turno 5
8	Se repite turno 5
9	Se repite turno 5

De la tabla 1, identificamos dos subrutinas principales: $mover(n)$ y $posible_triunfo$. La subrutina $mover(n)$, donde n es una de las 9 casillas del tablero, indica llevar la pieza a una posición. Por otra parte, $posible_triunfo$ es una sub-

rutina que formula un movimiento hacia cierta casilla en la cual el robot pueda ganar. Esencialmente, esta subrutina realiza su evaluación mediante una multiplicación de los estados en el tablero. Al tener una combinación [3, 3, 2], donde su multiplicación es “18”, entonces sabemos que el SCORBOT puede ganar, por lo tanto, la salida de la función será el número de casilla en la cual se puede generar el triunfo, si no es así arrojará como resultado un cero. En el caso de obtener un cero en la evaluación del triunfo del robot, esta subrutina es utilizada nuevamente, ahora para evaluar si el usuario “O” puede ganar. El procedimiento es similar al anterior pero en este caso la combinación buscada es [5, 5, 2] equivalente a “50” en la multiplicación. Esta subrutina también arroja como resultado el número de la casilla en la cual se puede generar la victoria del usuario. Puede notarse que el turno 4 requiere más evaluaciones que el resto de los turnos, debido a que en él se ha agregado la verificación de las casillas pares para evitar una derrota del robot, en el caso de que el usuario juegue con esas casillas menos ponderadas en el juego.

5. Robot manipulador: SCORBOT

El manipulador SCORBOT-ER 4U es un robot vertical con 5 grados de libertad y una pinza como elemento terminal para la manipulación de objetos. El robot incluye un controlador USB el cual cumple con varias funciones, destacando la función de actuar como la interfaz entre el manipulador y la computadora. Algunas características relevantes del manipulador son:

- Radio de operación máxima del robot es de 24,4”
- Apertura máxima de la pinza es de 2,6”
- Encoders ópticos en cada eje para posicionamiento.
- Carga máxima 1 kg
- Peso 10,8 kg
- Velocidad máxima de 600 mm/seg

En este proyecto se utiliza el controlador del robot para que ejecute las tareas de llevar la ficha de una posición “a” hacia una posición “b”. Donde “a” indica la posición donde el robot toma sus fichas de juego y “b” representa cada una de las posiciones del tablero. Además se tiene indicada una posición, que llamaremos posición de reposo, en la cual el robot espera a que el usuario realice su movimiento y llega a ella después de colocar una ficha en el tablero. Todas estas posiciones son guardadas, y después recuperadas desde un programa hecho en el lenguaje propio del robot para su envío al controlador a través del usb. Cabe mencionar que la decisión sobre cuál posición “a” hacia “b” debe realizarse, es directamente indicada por el algoritmo de juego.

5.1. Interfaz de comunicación entre módulos

Llamamos interfaz de comunicación al módulo de software que indica la secuencia de intervención de los distintos módulos del sistema. La interfaz utilizada

se muestra en la figura 4. El primero de los botones que presenta la interfaz es “Inicia”, el cual selecciona aleatoriamente si será el robot o el usuario quién inicie la partida. El segundo botón es “terminar turno” que indica al robot que el usuario ha realizado su movimiento y que puede salir de su posición de reposo, para realizar su siguiente jugada. Cada vez que el usuario indica el fin de su movimiento desde este segundo botón, la rutina del sistema de visión se encarga de capturar la imagen, procesarla y enviar el estado del tablero hacia el algoritmo de juego. El siguiente movimiento del robot es entonces calculado y enviado al controlador para que el SCORBOT realice el movimiento y termine su participación llegando a la posición de reposo. Finalmente el botón “Salir” de la interfaz da por terminado el juego, preparando todo el sistema para una nueva partida.



Figura 4. Interfaz de comunicación

6. Resultados

Una vez integradas las diferentes partes de nuestro sistema (sistema de visión, computadora, controlador y robot), se realizaron diversas jugadas a modo de prueba, observando detenidamente su comportamiento para identificar la existencia de irregularidades durante el desarrollo de la partida. A ejemplo, se extiende la jugada mostrada en la figura 5. Cada imagen representa uno de los turnos desarrollados en esta partida. En este caso, el robot tiene el turno de inicio, sin embargo, como se mencionó anteriormente cualquiera de los dos jugadores puede iniciar.

En este primer turno, el SCORBOT lleva la ficha a la casilla 1. En el turno 2 el usuario coloca su ficha en la casilla 9. El algoritmo analiza el estado del tablero antes de arrojar su posición al SCORBOT, quien tiene el tercer turno 3. Se verifica que el mejor resultado lo obtiene colocando en alguna esquina, por lo que se coloca en la casilla 3. En el turno 4, la persona coloca la ficha en la casilla 2, bloqueando el posible triunfo del robot. Durante el turno 5, el SCORBOT

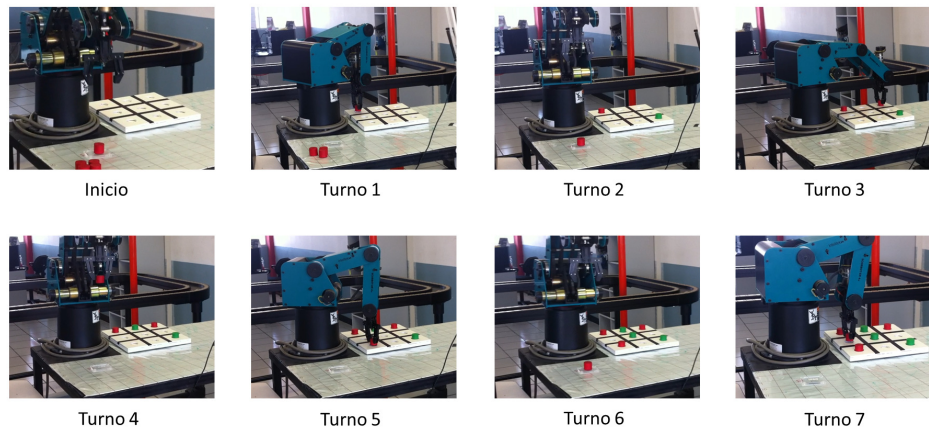


Figura 5. Desarrollo de una partida contra un usuario

lleva la ficha a la casilla 7, asegurando el triunfo con una jugada doble. La persona coloca la ficha a la casilla 5 durante el turno 6, bloqueando uno de los dos posibles triunfos del SCORBOT, sin embargo, el triunfo de éste último ya es inevitable en el turno 7, al colocar la ficha en la casilla 4. El video de esta partida puede ser visto en el enlace: http://www.youtube.com/watch?v=hxk_h2t8yso. Muchas otras partidas de “Tic-Tac-Toe” fueron realizadas, iniciando la partida tanto el SCORBOT como el usuario, obteniendo resultados satisfactorios en el desarrollo de las mismas, logrando al menos el empate del robot.

6.1. Análisis de desempeño del algoritmo

El equipo de cómputo utilizado para la implementación del algoritmo y sincronización con el resto del sistema fue una PC de escritorio con procesador Intel Core i7 2,2 GHz, Memoria cache L3 6 Mb, 8 Gb of RAM y sistema operativo windows XP para mejor compatibilidad con el controlador del robot. Desde la interfaz de Matlab, se realizaron unas mediciones del tiempo de ejecución ocupado por las partes esenciales del programa, (a nivel software) de lo cual encontramos:

- Duración de procesamiento de imagen: $41,2ms \pm 4,6ms$
- Duración de toma de decisión: $13,24\mu s \pm 7,9284\mu s$

Con esto garantizamos que toda la parte de código en Matlab, permita la ejecución de la tarea en tiempo real. Aún cuando este software no es de las mejores opciones para programación de sistemas tiempo real, comprobamos que para nuestra aplicación es suficiente. Con este resultado, sólo nos queda mencionar que las principales restricciones de desempeño se tienen a nivel hardware, esto es por el tiempo de adquisición de la imagen y el tiempo de respuesta mecánico

del SCORBOT, en promedio de 100 milisegundos, desde que el usuario indica su fin de turno hasta que el robot inicia su movimiento.

Por otra parte, una de las desventajas a considerar en nuestro sistema ocurre cuando el tablero o la cámara son movidos de su posición inicial seleccionada para los límites de las casillas. Dicho procedimiento, es una rutina que se realiza fuera de línea antes de iniciar la partida, por lo que es recomendable que una vez hecho esto, tanto el tablero como la cámara ya no se muevan de su posición. Sin embargo, de ser necesario, tomaría menos de un minuto realizar nuevamente esta tarea.

7. Conclusiones y perspectivas

En este documento se ha presentado un sistema integral de interacción hombre-máquina que desarrolla el clásico juego de “Tic-Tac-Toe” o “Tres en raya”. El sistema involucra diferentes áreas de aplicación como son la visión por computadora, programación de robots y de algoritmos de inteligencia artificial, permitiendo la realización de experimentos reales. La decisión de implementar el algoritmo presentado aquí y no otros, por ejemplo los que mencionamos basados en algoritmos genéticos, surgió de que este algoritmo nos permitiría una ejecución en tiempo real deseada para el sistema, obteniendo un buen compromiso entre tiempo de ejecución y eficiencia a la hora de formular un movimiento. Se pretende mejorar en el procesamiento de la imagen y hacer más robusto el sistema a cambios de iluminación, realizando un cambio de espacio de color, de RGB a CIE Lab. Por otra parte, se buscará la completa autonomía del sistema, ya sea aumentando la velocidad de adquisición de la cámara o algún otro tipo de sensor por el cual el robot identifique que el usuario ha terminado su turno y evitar dar clic sobre la interfaz. Sin embargo, nuestra principal perspectiva consiste en la puesta en marcha de este sistema, dado su carácter didáctico, en alguno de los centros de ciencias para niños de la región (los cuales cuentan con robots manipuladores en exposición), con el fin de que los niños y el público en general experimenten la interacción y el interés por los sistemas robóticos.

Agradecimientos Este trabajo fue parcialmente soportado por la Universidad de Guanajuato mediante el proyecto titulado “Sistema de Identificación de objetos mediante los atributos del color y la textura utilizando arquitectura reconfigurable con FPGA” con número 000175/11.

Referencias

1. Aliane, N., Bemposta, S.: Checkers playing robot: A didactic project. *Latin America Transactions, IEEE (Revista IEEE America Latina)* 9(5), 821–826 (Sep 2011)
2. Bhatt, A., Varshney, P., Deb, K.: In search of no-loss strategies for the game of tic-tac-toe using a customized genetic algorithm. In: *Proceedings of the 10th annual conference on Genetic and evolutionary computation*. pp. 889–896. GECCO '08, ACM, New York, NY, USA (2008), <http://doi.acm.org/10.1145/1389095.1389269>

3. Bicchi, A., Caiti, A., Pallottino, L., Tonietti, G.: Online robotic experiments for tele-education at the university of pisa. *Journal of Robotic Systems* 22(4), 217–230 (Apr 2005), <http://onlinelibrary.wiley.com/doi/10.1002/rob.20061/abstract>
4. Esposito, J.M., Wick, C., Knowles, K.: Matlab toolbox for the intelitek scorbot: An open source robotics education library. In: American Society of Engineering Education Annual Conference (2011)
5. Hochmuth, G., Koza, J.: On the genetic evolution of a perfect Tic-Tac-Toe strategy. In: Genetic Algorithms and Genetic Programming at Stanford 2003, pp. 75–82. Stanford University (2003)
6. Jara, C.A., Candelas, F.A., Torres, F.: Virtual and remote laboratory for robotics e-learning. In: 18th European Symposium on Computer Aided Process Engineering. vol. 25, pp. 1193–1198. Elsevier (2008), <http://rua.ua.es/dspace/handle/10045/10159>
7. Kay, J.S.: From mad libs to tic tac toe: Using robots and game programming as a theme in an introduction to programming course for Non-Majors. In: Twenty-Second International FLAIRS Conference (Mar 2009), <http://aaai.org/ocs/index.php/FLAIRS/2009/paper/viewPaper/126>
8. Rich, E., Kevin, K.: *Inteligencia Artificial*. McGrawHill (1994)

Seguridad informática

Hacia la detección de impostores mediante grafos de navegación

J. Benito Camiña, Carlos Hernández-Gracidas y Raúl Monroy

ITESM CEM, Atizapán, Estado de México, México
{a00965049, carloshg, raulm}@itesm.mx

Resumen Dado que la información es un bien valioso, la detección de impostores es vital en los sistemas computacionales. El auge en el estudio de la detección de impostores se dio principalmente a partir del trabajo seminal del grupo de Schonlau, donde perfilan a los usuarios con base en los comandos que utilizan, produciendo un conjunto de datos que, por años, ha sido el estándar para comparar métodos de detección de impostores. Sin embargo, el desempeño de estos métodos no es concluyente, y, como resultado, se han investigado fuentes alternativas de información. En este artículo mostramos cómo se construyó un conjunto de datos conteniendo perfiles de usuarios a partir de sus accesos a objetos. Complementariamente, se describe una serie de ataques simulados. Finalmente, proponemos el uso de un vector de atributos como una forma de caracterizar al usuario y diferenciarlo de un impostor, proponiendo posibles experimentos sobre este conjunto de datos.

Palabras clave: Detección de intrusiones, conjunto de datos, grafos de navegación.

1. Introducción

Para el mundo actual, donde se acepta que la información es un bien extremadamente valioso, la detección a tiempo cuando nuestra computadora (sesión) está siendo ilegalmente utilizada por un intruso, conocido como impostor, es vital. El propósito de un impostor es poner en riesgo al usuario de una computadora, tan rápido como sea posible, mientras se hace pasar por el usuario para evitar ser detectado.

La detección de intrusos ha sido estudiada activamente a partir del trabajo seminal de Schonlau et al. [1], que sugiere que, para perfilar a un usuario, debe ser suficiente con modelar el historial de los comandos (sin parámetros) que introduce mientras tiene una sesión UNIX activa. Para tal propósito, Schonlau et al. desarrollaron un conjunto de datos de impostores, comúnmente conocido como SEA [2], que ha sido utilizado como un estándar *de facto* para construir, validar y comparar una gran cantidad de métodos de detección (véanse, por ejemplo, [3,4]). Sin embargo, el desempeño de métodos basados en SEA no puede decirse que sea abrumador. Así, SEA ha sido enriquecido para considerar información

adicional (en particular, argumentos de comandos); véase por ejemplo, [6]. También como resultado, la investigación en detección de impostores ha optado por el uso de fuentes de actividad alternativas en un intento por caracterizar mejor el comportamiento del usuario. Ejemplos de fuentes alternativas de información incluyen el comportamiento del usuario en un entorno gráfico [14], y el uso de aplicaciones como el empleo de un sistema de administración de documentos [8].

En este artículo, afirmamos que para caracterizar mejor el comportamiento del usuario, es necesario considerar cómo y qué recorre el usuario mientras trabaja en su sistema de archivos. Después de todo, el sistema de archivos de un usuario es una representación abstracta del usuario, al menos en cuanto al sistema de archivos se refiere. Crucial para nuestro enfoque es una estructura, que llamamos la estructura de navegación del usuario y que ya se presentó con resultados preliminares en [16], representando la navegación de un usuario a través de su sistema de archivos. Una estructura de navegación contiene información del sistema de archivos del usuario. En particular, contiene los objetos más recientemente visitados en el sistema de archivos; también contiene información tanto acerca de cómo el usuario estructura su sistema de archivos, como de la manera en que usa y recorre tal directorio. Mostraremos que usar la estructura de navegación de un usuario obedece a un doble propósito. Primero, la estructura de navegación permite entender mejor el comportamiento de un usuario dado. Segundo, y más importante, permite construir un perfil de usuario adecuado para la detección de impostores. Aún cuando reconocemos que es necesaria la combinación de varios tipos de actividades, nuestro planteamiento pretende demostrar que la navegación del sistema de archivos es determinante para conseguir un perfilado preciso del usuario.

Perspectiva general del artículo El resto de este artículo está organizado de la siguiente forma. Primero, en 2 mostraremos las limitaciones, en general, de los métodos de detección de impostores. En particular, argumentaremos que, por una parte, se desempeñan bastante pobremente, aún cuando no son sujetos a acciones propias de intrusos; y que, por otra parte, son difíciles de implementar, porque la información de actividad elegida para perfilar a un usuario es, por naturaleza, poco densa. Segundo, en 3, se definen brevemente las estructuras de navegación del usuario en que basamos nuestra representación de la información del usuario, y hacemos referencia a [16], donde se definen con mayor amplitud, junto con las operaciones que toman. Tercero, en 4, bosquejaremos el conjunto de datos que hemos recolectado, el cual se divide en datos del usuario y datos de los ataques realizados. Describimos la manera en que se realizó un prefiltrado de las bitácoras con el fin de conservar únicamente la información relevante y deshacerse del ruido proveniente de los registros originales. Cuarto, en 5, proponemos una representación vectorial formada por atributos extraídos de los registros, y que consideramos pueden ser de utilidad para caracterizar al usuario. Estos vectores son obtenidos dividiendo los registros en ventanas de tamaño fijo y se proponen como una fuente de información adecuada para la creación del perfil del usuario y la detección de impostores. Quinto, en 6, se propone una serie de experimentos que pueden realizarse a partir del conjunto de datos y la repre-

sentación de los mismos que se ha propuesto. Las ideas de investigación futura tienen el propósito de sugerir algunas técnicas que permitan explotar la información recabada. Finalmente, en 7, reportaremos las conclusiones que hemos obtenido hasta el momento.

2. Mecanismos de detección de impostores: perspectiva general

La investigación en detección de impostores ha sido muy prolífica. Las limitaciones de espacio nos impiden dar un panorama general razonable, o una comparación justa de estos mecanismos (aún si nos limitamos a los más representativos). Debemos confinarnos a dar una perspectiva general de unos cuantos, remitiendo al lector a [9] para un estudio detallado.

Un detector de impostores es *global* si, para determinar la presencia de un impostor, usa tanto el perfil del usuario que está siendo protegido, como el perfil de los colegas de dicho usuario. Sino, es *local*. Igualmente, un detector de impostores es *basado en temporalidad* si, para la formación del perfil del usuario, considera tanto acciones individuales como la relaciones entre ella; por ejemplo, secuencias de acciones. Sino, es *basado en frecuencias*. Dado que los métodos globales y temporales se encuentran mejor informados, usualmente son más precisos que los locales y los basados en frecuencia; sin embargo, requieren más información y esfuerzo computacional, y son inaplicables en algunos contextos.

Gramáticas personalizadas [4] es un método de detección global, basado en temporalidad, que perfila el comportamiento normal del usuario usando las secuencias de acciones del usuario únicas y las más repetitivas. Para identificar tales secuencias de acciones dentro del historial del usuario y respecto a los registros de auditoría de otros usuarios, las gramáticas personalizadas aplican *sequitur* [10], un método para inferir jerarquías composicionales a partir de cadenas.

Naïve Bayes [11,6,12] asume que las acciones de un usuario son independientes entre sí. Así, la probabilidad de que una acción c haya sido originada por un usuario u , $Pr_u(c)$ está dada por $\frac{f_{uc} + \alpha}{n_u + \alpha \times K}$, donde f_{uc} es el número de veces que el usuario u empleó c en el registro de acciones del usuario, n_u es la longitud del registro de u , K es el número total de acciones distintas, y donde $0 < \alpha \ll 1$ previene que el valor de $Pr_u(c)$ sea cero. Así, naïve Bayes es basado en frecuencias y puede ser tanto local como global, con desempeño similar [12]. Para evaluar una sesión de prueba s , en la que el usuario u presuntamente participó, la probabilidad acumulativa de s es comparada ya sea contra la probabilidad de que s haya sido producido por alguien más [11] o contra un umbral definido por el usuario [12].

Sequitur-luego-HMM [5] es un detector de impostores local y temporal, que caracteriza el comportamiento del usuario en términos de las dependencias temporales entre secuencias de acciones más frecuentes. Este método aplica primero *sequitur* a un historial de acciones de usuario dado. Luego, la principal regla de producción del resultado de la gramática generada por *sequitur* es empleada para entrenar un modelo oculto de Markov (HMM, por sus siglas en inglés).

Consecuentemente, este modelo no solo considera dependencias temporales entre acciones, debido a *sequitur*, sino que también considera dependencias temporales entre secuencias, debido al HMM.

Otras fuentes de información Dado que el uso de información como la contenida en la base de datos SEA [2] para el perfilado y detección de impostores parece no ser suficiente como fuente de información, recientemente la investigación en esta área ha buscado fuentes alternativas de actividad del usuario para perfilar mejor su comportamiento. En [8] se ha experimentado con otro tipo de fuente como lo es la interacción del usuario con aplicaciones específicas (por ejemplo, el uso de un sistema de administración de documentos). En [14] se presenta trabajo con redes neuronales artificiales (ANNs) en el aprendizaje de características del usuario en un sistema gráfico. Las características aprendidas son el número de veces que una acción específica es observada, el número de veces que se presionan teclas de control, y el número de veces que se ejecutan ciertos procesos. Finalmente, en [15] los comandos del usuario se agrupan, y luego se enlazan en forma de tipos de actividades, tales como recopilación de información, búsqueda de información, comunicaciones, etc., con el objetivo de capturar la intención del usuario, probando que es fácil detectar a un impostor dado que es muy posible que éste exhiba un comportamiento de búsqueda propios de alguien que desconoce la computadora de la víctima. Aunque la idea clave detrás de este trabajo es interesante, actualmente el etiquetado de acciones requiere de intervención humana.

El desempeño de los métodos descritos (y otros) se encuentra, sin embargo, distante de ser concluyente. Esto ha sido probado en [13], donde reportan que la mayoría de estos métodos, cuando operan al 95 % de tasa de detección, producen más del 40 % de falsas alarmas. Los detectores de impostores no han sido completa o confiablemente probados. Esto es porque no han sido probados bajo condiciones críticas, donde se requiere distinguir a un intruso que realmente intenta hacerse pasar por un usuario (ver también [13]). A partir de esto, se deduce que estos métodos tampoco han sido comparados exhaustivamente.

3. Estructuras de navegación

Para capturar cómo recorre un usuario su sistema de archivos, empleamos una estructura de navegación. Una estructura de navegación contiene información tanto acerca de cómo un usuario estructura su sistema de archivos, como de la manera en que la emplea y recorre. Los objetos en una estructura de navegación han sido visitados por el usuario, ya sea directamente, mediante un recorrido de directorios, o indirectamente, mediante una aplicación del usuario.

Por limitaciones de espacio, no es posible explicar de manera amplia la estructura de navegación empleada; sin embargo, ésta ya fue explicada y detallada en un trabajo previo [16], y sugerimos revisarlo para su correcto y completo entendimiento. A continuación, nos acotaremos a dar una idea general de las características de la estructura de navegación.

Una estructura de navegación está compuesta de dos grafos: un grafo de accesos y un grafo de directorios. Un *grafo de accesos* contiene un registro de los objetos del sistema de archivos que han sido visitados más recientemente, así como el orden (de hecho, un salto de ruta) que el usuario ha seguido para visitarlos. Un *grafo de directorios* es un subgrafo propio del sistema de archivos del usuario; es, de hecho, una arborescencia, con un vértice distinguido, llamado la *raíz*, y denotado por “/”. A partir de esta arborescencia es posible definir el concepto de *profundidad*, que representa la cantidad de desplazamientos, partiendo de la *raíz*, que deben realizarse para alcanzar un objeto dado, tomando la profundidad de la *raíz* como 0 y considerando que dicha profundidad se incrementa (o correspondientemente, decrementa) en 1 con cada desplazamiento hacia abajo (o correspondientemente, hacia arriba) de la arborescencia.

Mantenimiento de la Estructura de Navegación Para mantener la estructura de navegación de un usuario de un tamaño manejable, es posible y necesario aplicar un algoritmo de reemplazo que permita actualizar la estructura de navegación cuando sea necesario; por ejemplo, después de que el usuario ha movido o removido un objeto del sistema de archivos. Esto permitirá manejar adecuadamente el concepto de *objetos nuevo*, en términos de accesos o secuencias de accesos en un período de tiempo, pues después de un tiempo determinado sin utilizarse, ciertos objetos podrían ser removidos de la estructura como parte de su mantenimiento y volver a considerarse como “nuevos” si se acceden otra vez.

4. El conjunto de datos

Es necesario, para una correcta experimentación, contar con un conjunto de bitácoras confiable, tanto de usuarios como de ataques. El conjunto de datos se divide en datos de usuarios válidos y datos de ataques realizados, por lo que la tarea de obtención de los mismos también fue dividida en dos etapas.

4.1. Creación del conjunto de datos de usuarios

Para la obtención del conjunto de datos de usuarios se reclutaron voluntarios, usuarios frecuentes de (alguna versión de) el sistema Windows. Con su ayuda, y bajo previa firma de un convenio de colaboración y protección de datos, configuramos el sistema de auditoría, el cual es una herramienta ya existente en todas las versiones de Windows con las que se experimentó. Actualmente se cuenta con un total de 20 usuarios, aunque, a la fecha, se ha recabado información de ataques para 14 de ellos, faltando por simularse los ataques para los 6 usuarios restantes. El perfil de los 14 usuarios para los que se dispone de registros completos se muestra en la Tabla 1, habiéndose capturado en períodos que varían de poco más de 15 días hasta 3 meses.

Se elaboró una encuesta con el objetivo de contar con la mayor información posible de los usuarios y, así, estas características puedan ser empleadas para clasificarlos. De aquí, se observa que nuestra base cuenta con usuarios de perfiles heterogéneos (pero que pueden separarse en unas cuantas clases), en

Tabla 1. Perfil de los 14 usuarios para los que se cuenta con un registro completo. Se buscó que los perfiles fueran variados, obteniéndose registros de distintas versiones de Windows, de entre 17 y 67 días de captura.

Usuario	Rol	Sistema Operativo	Fecha de inicio	Fecha final
1-2	Administrador General	Windows XP	14/02/2012	24/03/2012
3	Encargado de Sistemas	Windows XP	14/02/2012	24/03/2012
4	Contador	Windows XP	14/02/2012	24/03/2012
5-9	Secretarías de Empresas	Windows XP	14/02/2012	24/03/2012
10	Encargado de Entregas	Windows Vista	14/02/2012	24/03/2012
11	Secretaría de Universidad	Windows XP	21/03/2012	13/04/2012
12	Programadora de Sistemas	Windows 7	16/01/2012	23/03/2012
13	Estudiante Universitaria	Windows Vista	22/01/2012	23/03/2012
14	Gerente de Ventas	Windows 7	16/12/2011	28/03/2012

términos de: conocimiento del sistema operativo, perfil profesional, frecuencia de uso del sistema, nivel de organización, etc. Nuestra muestra permite obtener conclusiones estadísticamente confiable respecto a los factores que influyen en una mejor detección de impostores, y, contando con la información adicional que se ha mencionado, esperamos poder obtener ciertos indicios de cómo el tipo de usuario influye en la detección de impostores (por ejemplo, una hipótesis que se pretende probar es que aquellos usuarios que son más organizados serán más fáciles de defender de una intrusión que aquellos que no lo son).

4.2. Creación del conjunto de datos de ataques

Para la creación y posterior simulación de ataques impostores se recurrió a una encuesta diseñada por el equipo y aplicada a casi 50 personas. En la encuesta, los participantes actuaron bajo un escenario hipotético en el que podrían hacer uso, no autorizado, de una computadora desatendida, por 5 minutos. Recopilamos el tipo de información que, como impostores, los encuestados buscarían, además del tipo de búsqueda a emplear, y la forma en que intentarían hacerse de la información. Así, diseñamos 3 tipos de ataques: básico, intermedio y avanzado (ver Tabla 2). El objetivo de estos ataques es comprometer información relevante del usuario, para lo cual, el atacante interactúa con la computadora navegando en el sistema de archivos. Cabe resaltar que cada ataque se implementó mediante un conjunto de pasos, de modo que su simulación es uniforme y controlada. Tras la simulación, recopilamos bitácoras, que pueden ser usadas para evaluar la efectividad de los mecanismos de detección de impostores. Tomando en cuenta que se realizaron tres ataques diferentes por cada usuario válido, se obtuvo un total de 42 ataques.

4.3. Prefiltrado de bitácoras

Dada la diversidad de los datos, por las distintas versiones de sistema operativo, los diferentes tipos de usuarios, los distintos tiempos de registro, las variaciones en la cantidad de registros, etc., y porque el sistema de auditoría genera información en demasía, los prefiltramos, obteniendo información consistente,

Tabla 2. Características principales de los ataques realizados a los 14 usuarios. Todos los ataques son restringidos a una duración máxima de 5 minutos.

Ataque	Características
1. Básico	Apertura manual de carpetas y robo de información copiando manualmente o tomando fotografías.
2. Intermedio	Búsqueda de archivos especiales con el buscador de Windows y copiado de los archivos a una memoria USB.
3. Avanzado	Ejecución de un script .bat en una memoria USB que se encarga de copiar automáticamente archivos a la memoria.

sin importar su fuente, para quedarse con un listado de las acciones ocurridas en la computadora del usuario, únicamente en las carpetas seleccionadas para su monitoreo y que contuviera información exclusivamente del objeto al que se accedió, así como el momento en que fue accedido. Contar con este prefiltrado permite una creación del modelo más eficiente y reduce considerablemente el ruido proveniente de los datos recabados. Al final de este proceso, la información que es de nuestro interés es la siguiente:

$ID_acceso|Fecha|Hora|Tiempo_transcurrido|Profundidad|Ruta_acceso$

donde ID_acceso representa el identificador del número de acceso realizado por el usuario, permitiendo llevar un orden consecutivo de los accesos. $Fecha$ contiene la fecha del acceso. $Hora$ contiene la hora en formato de 24 horas. $Tiempo_transcurrido$ es un campo especial que contiene el tiempo que ha transcurrido (medido en segundos) desde una fecha predeterminada (en nuestro caso, desde el 01/01/2011) hasta el momento del acceso, y que nos permitirá hacer cálculos como el tiempo entre accesos de una manera sencilla. $Profundidad$ contiene la profundidad de la ruta. Y, finalmente, $Ruta_acceso$ contiene la ruta completa del objeto en el sistema de archivos.

5. Creación del modelo de representación de los datos

Después de realizar el filtrado es necesaria la división de la bitácora en ventanas y la obtención de atributos (para cada una de esas ventanas) que servirán después para la detección de intrusos. Para ello, se empleará un modelo de representación basado en grafos de navegación que refleja la interacción del usuario en el sistema de archivos. Posteriormente, un conjunto de atributos es extraído a partir de la información contenida en las ventanas, para así formar vectores de atributos (estos vectores deben ser calculados tanto para los datos del usuario como para los del atacante). El modelo de representación ya fue propuesto en un trabajo previo con resultados alentadores [16], donde se empleó naïve Bayes junto con ventanas definidas en términos del número de acciones (a diferencia del presente trabajo, donde se propone todo un vector de atributos y la definición de ventanas en términos de tiempo, como se verá más adelante). Los vectores de atributos propuestos tienen la siguiente estructura:

La información contenida en estos vectores es: i) Accesos. Se obtiene el número total de accesos en la ventana (T) así como el número de accesos nuevos

Tabla 3. Atributos extraídos de los grafos de navegación. Este es el vector que describe los atributos obtenidos del grafo de navegación, que a su vez es dividido en ventanas, con los que se pretende realizar la identificación de impostores.

Accesos		Profundidad					Tiempo entre accesos					Características del grafo					Distancia en recorrido entre accesos					Probabilidad				
T	N	\bar{x}	\tilde{x}	M_o	S^2	S	R_e	\bar{x}	\tilde{x}	M_o	S^2	S	R_e	$ V $	$ E_d $	$ E_u $	$\frac{ V }{ E_d }$	$\frac{ V }{ E_u }$	\bar{x}	\tilde{x}	M_o	S^2	S	R_e	NB_v	NB_e

(N); ii) Profundidad. Con la profundidad a que se encuentra cada objeto accediendo en la ventana, se obtienen media (\bar{x}), mediana (\tilde{x}), moda (M_o), varianza (S^2), desviación estándar (S) y rango muestral (R_e) dentro de la ventana; iii) Tiempo entre accesos. Utilizando el tiempo del campo *Tiempo_transcurrido* se puede calcular el tiempo que transcurre entre un acceso y otro dentro de la ventana, y con estos tiempos también es posible calcular media, mediana, moda, varianza, desviación estándar y rango muestral dentro de la ventana; iv) Características del grafo. En cuanto a las características del grafo, se busca obtener el número total de vértices ($|V|$), el número total de aristas dirigidas ($|E_d|$), el número total de aristas no dirigidas ($|E_u|$), la proporción de vértices entre aristas dirigidas ($\frac{|V|}{|E_d|}$) y la proporción de vértices entre aristas no dirigidas ($\frac{|V|}{|E_u|}$); v) Distancia en recorrido entre accesos. También se calcularán distancias entre los accesos, esto basándose en la distancia que hay de un nodo a otro dentro del grafo, y se calcularán las mismas medidas estadísticas, que son: media, mediana, moda, varianza, desviación estándar y rango muestral. vi) Probabilidad. Se calcularán dos formas distintas de naïve Bayes, una basándose en la probabilidad del acceso a los vértices (NB_v) y otra basándose en la probabilidad de acceso a las aristas (NB_e).

Por otro lado, como se mencionó, se pretende perfilar al usuario y detectar a un impostor a partir de la división de la información en ventanas definidas en términos de un tiempo fijo. Así, los registros se analizarán por ventanas divididas por un tiempo t , donde t es un tiempo (en segundos) constante durante cada análisis. Uno de los objetivos de estos experimentos es realizar pruebas usando distintos tamaños de ventana (por ejemplo, 30, 60 y 90 segundos) para encontrar cuál es más apropiada para la detección de intrusos, pues, se espera que las variaciones en la cantidad de objetos presentes en una ventana sirvan como indicios de la actividad de un impostor en el sistema.

6. Experimentos propuestos

A continuación describimos una serie de experimentos que se realizarán con el conjunto de datos de usuarios y atacantes con que se cuenta. Cabe mencionar que, aunque estos experimentos no se han realizado, el proceso para la realización de los mismos se encuentra ya avanzado, por lo que se espera contar con resultados preliminares en un corto plazo.

Recordando que se cuenta con una bitácora ya filtrada por cada usuario, en la etapa experimental, para generar el perfil del usuario se irán analizando las ventanas generando los vectores mencionados en la Sección 5. El proceso

de creación del perfil se realizará individualmente por cada usuario, por lo que, para k usuarios se tendrán k perfiles completamente independientes, que se espera capturen los patrones de comportamiento específicos a cada uno de ellos. Los datos obtenidos se emplearán para actualizar los grafos de directorios y de accesos para el usuario; de esta forma, es posible generar el perfil del usuario reflejado en un árbol, así como obtener los vectores que representen su comportamiento.

Es necesario tomar en cuenta el hecho de que es necesario contar con un período de ajuste del sistema (aquél donde los datos obtenidos deban ser considerados como poco fiables), pues, por ejemplo, ya que al inicio de los registros habrá un gran número de accesos nuevos a objetos (en realidad, desde cierto punto de vista, todos los accesos serán nuevos, puesto que no se tendrá registro de que se hayan realizado previamente). Esto lleva a la necesidad de desechar los primeros registros (calculamos que el porcentaje de datos iniciales desechados no sobrepase el 5%), para evitar capturar ruido al perfilar al usuario. Esta información, sin embargo, será desechada parcialmente, pues, aunque no se considerarán algunos aspectos de ella para perfilar al usuario, sí es necesario actualizar los árboles que se generan con información de estos accesos. En la Figura 1 se ve una posible forma en que podrían distribuirse los datos de las bitácoras de los usuarios.

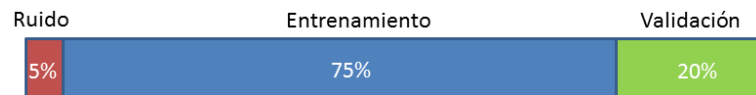


Figura 1. Distribución sugerida para los datos. El 5% inicial se descartaría para la creación del perfil del usuario, aunque sí se emplearía para actualizar los grafos de directorios y accesos; el 75% siguiente se emplearía para crear el perfil del usuario; y, finalmente, el último 20% se emplearía para validar el modelo obtenido.

Aunque sugerimos descartar el 5% inicial, emplear el 75% siguiente para entrenamiento y usar el último 20% para la validación del modelo, este esquema no se propone como algo rígido, puesto que algunos aspectos que se desean evaluar tienen relación directa con estos porcentajes y, por ende, requieren experimentar variándolos. Algunas de las preguntas que surgen a este respecto son:

- ¿Qué porcentaje de datos es fiable descartar considerándolo como ruido?
- ¿Qué porcentaje de datos es necesario utilizar para el perfilado del usuario?
- ¿Qué porcentaje de datos basta para validar correctamente el modelo?
- ¿Estos porcentajes son fijos o varían, dependiendo de las características del usuario como qué tan organizado es, cuánta información se recabó, durante qué tiempo se recabó, etc.?

Una vez contando con los vectores que representan los grafos de navegación, se propone sacar provecho de ellos mediante la aplicación de métodos inteligentes

para aprender las características del usuario. Dado que, para los ataques realizados se cuenta también con una bitácora, ésta será también preprocesada y será posible obtener una representación vectorial de estos ataques. Con esta información será posible evaluar si una nueva ventana corresponde a la actividad del usuario válido o, por el contrario, se trata de un potencial intruso. Para tal evaluación se sugieren como opciones el uso de ANNs, SVMs, K vecinos más cercanos (KNN) y regresión lineal localmente ponderada (LWLR). Un esquema general de la clasificación (incluyendo el preprocesado y la obtención de los vectores), se muestra en la Figura 2.

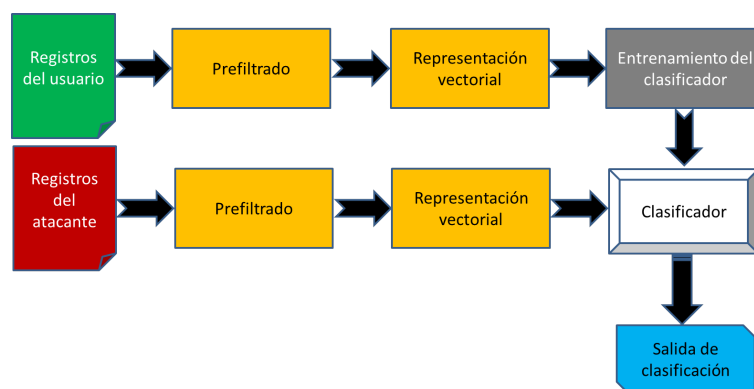


Figura 2. Diagrama del esquema de detección de impostores propuesto, basado en grafos de navegación.

El uso de varios métodos para clasificación haría posible evaluar el desempeño de los distintos algoritmos en términos de falsos positivos (el caso cuando una ventana del usuario sea clasificada erróneamente como del atacante) y falsos negativos (el caso opuesto, cuando una ventana del atacante sea clasificada erróneamente como del usuario), principalmente, además de que permitiría sacar conclusiones acerca de qué método se adapta mejor al tipo de datos con que se trabaja. Se propone que estos clasificadores sean entrenados mediante el enfoque de una clase, es decir, proporcionándole al clasificador la información del usuario para el entrenamiento, y esperando que a partir de ella pueda detectar si una nueva entrada corresponde al mismo usuario o no, en cuyo caso se determinaría que se trata de un impostor.

7. Conclusiones y trabajo futuro

La estructura de navegación que se emplea en este trabajo para caracterizar al usuario respecto a su interacción con el sistema de archivos ya la hemos propuesto anteriormente [16]. En dicho trabajo se presentaron resultados preliminares que daban una idea del potencial de tal estructura.

En este trabajo, hemos ido más allá al corregir las limitaciones antes propuestas al extender los datos del usuario a 14 personas con registros de su interacción con el sistema de varios días. Los participantes del experimento son heterogéneos en términos de su perfil de usuario, lo que permitirá sacar mejores conclusiones de los experimentos que se realicen.

La metodología seguida para los ataques pretende ayudar a homogeneizar los mismos, de manera que las pruebas con los 3 ataques realizados por usuario (42 en total), sean de utilidad para evaluar diversas técnicas de detección de impostores. Se ha propuesto una representación de los grafos mediante su división en ventanas de tiempo y su posterior representación en vectores de atributos, que sean empleados para entrenar métodos de clasificación de una clase.

Con la información que se cuenta, es posible comenzar a probar los métodos de detección propuestos en este documento. Se espera que con los atributos que se extraen y haciendo uso de los grafos de accesos y de directorios se obtengan mejores resultados que en trabajos previos, principalmente en términos de porcentajes bajos de falsos positivos y falsos negativos.

A mediano plazo se planea agregar más campos a la estructura de navegación así como nuevos atributos al vector que, esperamos, puedan caracterizar mejor a un usuario. También se tiene contemplado utilizar otros tipos de clasificadores de una sola clase para encontrar cual es el más conveniente para la detección de intrusos. Finalmente, se espera contar con nuevos usuarios con perfiles distintos para poder generar conclusiones basadas en el tipo de usuario y/o sistema operativo utilizado.

Agradecimientos. Este proyecto fue apoyado parcialmente por CONACyT mediante una beca de doctorado y una beca de estancia posdoctoral. Agradecemos por su contribución en la extracción de los atributos de las ventanas a vectores y la obtención de los datos para la normalización de los vectores al estudiante Eduardo Murillo González.

Referencias

- [1] Schonlau, M., DuMouchel, W., Ju, W., Karr, A., Theus, M., Vardi, Y.: Computer intrusion: Detecting Masquerades. *Statistical Science* 16,58–74, (2001)
- [2] Schonlau, M.: Masquerading user data. <http://www.schonlau.net> (2008)
- [3] Maxion, R.A., Townsend, T.N.: Masquerade detection augmented with error analysis. *IEEE Transactions on Reliability* 53, 124–147 (2004)
- [4] Latendresse, M.: Masquerade detection via customized grammars. In Julish, K., Kruegel, C., eds.: *Proceedings of the Second International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, DIMVA 2005*. Volume 3548 of *Lecture Notes in Computer Science.*, Springer, 141–159, (2005)
- [5] Posadas, R., Mex-Perera, C., Monroy, R., Nolasco-Flores, J.: Hybrid method for detecting masqueraders using session folding and hidden markov models. In: *Proceedings of the 5th Mexican International Conference on Artificial Intelligence: Advances in Artificial Intelligence*. Volume 4293 of *Lecture Notes in Computer Science.*, Springer, 622–631 (2006)

- [6] Maxion, R.A.: Masquerade detection using enriched command lines. In: Proceedings of the International Conference on Dependable Systems and Networks, DSN'03, San Francisco, CA, USA, IEEE Computer Society Press, 5–14 (2003)
- [8] Sankaranarayanan, V., Pramanik, S., Upadhyaya, S.: Detecting masquerading users in a document management system. In: Proceedings of the IEEE International Conference on Communications, ICC'06. Volume 5., IEEE Computer Society Press, 2296–2301, (2006)
- [9] Salem, M.B., Hershkop, S., Stolfo, S.J.: A survey of insider attack detection research. In Stolfo, S.J., Bellovin, S.M., Hershkop, S., Keromytis, A., Sinclair, S., Smith, S.W., eds.: *Insider Attack and Cyber Security: Beyond the Hacker. Advances in Information Security*. Springer, 69–90, (2008)
- [10] Nevill-Manning, C.G., Witten I.H.: Identifying hierarchical structure in sequences: a linear-time algorithm. *Journal of Artificial Intelligence Research*, 67–82, (1997)
- [11] Maxion, R.A., Townsend, T.N.: Masquerade detection using truncated command lines. In: Proceedings of the International Conference on Dependable Systems & Networks, Washington, DC, IEEE Computer Society Press, 219–228, (2002)
- [12] Wang, K., Stolfo, S.: One-class training for masquerade detection. In: Proceedings of the 3rd IEEE Workshop on Data Mining for Computer Security, IEEE (2003)
- [13] Razo-Zapata, I., Mex-Perera, C., Monroy, R.: Masquerade attacks based on user's profile. Submitted to *Journal of Systems and Software* (2011)
- [14] Imsand, E.S., Garrett, D., Hamilton, J.A.: User identification using gui manipulation patterns and artificial neural networks. In: *Computational Intelligence in Cyber Security*, 130–135 (2009)
- [15] Ben-Salem, M., Stolfo, S.: Modeling user search behavior for masquerade detection. In: *Research Advances in Intrusion Detection, RAID'2011, Lecture Notes in Computer Science*, in press. Springer (2011)
- [16] Camiña, B., Monroy, R., Trejo, L.A., Sánchez, E.: Towards building a masquerade detection method based on user file system navigation. In: Proceedings of the Mexican International Conference on Artificial Intelligence, 174–186, (2011)

Un nuevo esquema para la detección de ataques en redes inalámbricas

Jorge Vázquez, Raúl Monroy y Luis A. Trejo

Departamento de Ciencias Computacionales,
Instituto Tecnológico y de Estudios Superiores de Monterrey,
Campus Estado de México, México
{A01165550,raulm,ltrejo}@itesm.mx

Resumen En este trabajo presentamos una metodología para la detección de anomalías en redes inalámbricas IEEE 802.11. Señalamos las dos principales vulnerabilidades del protocolo, la de identidad y la de acceso al medio. Nuestra metodología se basa en la caracterización de secuencias de tramas de capa 2, a estas secuencias las llamamos conversaciones. Para definir un conversación realizamos una abstracción de una trama a un símbolo, mediante el uso de técnicas de mapas auto organizables, k-medias y el algoritmo C4.5. Con estos elementos construimos un compresor. El cual genera secuencias más cortas que esperamos representen eficientemente comportamiento normal. Nuestro análisis refleja un porcentaje de reducción cercano al 92 %. Finalmente, proponemos una arquitectura en la que se usarán modelos ocultos de *Markov* con el fin de obtener un modelo de comportamiento normal y que pueda determinar la similitud o diferencia entre dos conjuntos de datos.

1. Introducción

El uso creciente de las redes inalámbricas basadas en el protocolo IEEE 802.11 ha incrementado el problema de accesos no autorizados, provocando un uso inadecuado de los recursos que provee un sistema de red. Este tipo de intrusiones se definen como cualquier actividad maliciosa dirigida a los servicios que la red provee, y se ve como una secuencia de acciones que alteran su estado y comprometen su seguridad. Para proteger los sistemas de red de estas actividades maliciosas se han desarrollado los sistemas de detección de intrusiones o IDS, por sus siglas en inglés *Intrusion Detection Systems*. Los IDS están encargados de monitorear eventos para el descubrimiento oportuno de actividades que ponen en riesgo la integridad, disponibilidad y confidencialidad de una red inalámbrica. En este trabajo nos interesa desarrollar un detector de intrusiones que modele la interacción entre una computadora y la red, identificando desviaciones en el comportamiento normal de dicha interacción. Es de especial interés la detección de ataques de denegación de servicio o DoS, por sus siglas en inglés *Denial of Service*. Estos ataques consisten, por ejemplo, en saturar un sistema con peticiones falsas de tal manera que los servicios o recursos del sistema atacado no estén disponibles para los usuarios genuinos que lo soliciten.

2. El protocolo IEEE 802.11

El protocolo IEEE 802.11 se centra en la capa de control de acceso al medio o MAC, por sus siglas en inglés *Media Access Control*, del modelo OSI, específicamente en el control de enlace lógico. La manera en que una estación o STA, acrónimo del inglés *Station*, obtiene acceso a una red mediante un punto de acceso o AP, por sus siglas en inglés *Access Point*, puede verse como una máquina de estados finitos o FSM, por sus siglas en inglés *Finite State Machine*. Las tramas que son intercambiadas son de tres tipos: de control, de mantenimiento y de datos. Cada tipo de trama, a su vez, se divide en sub-tipos de tramas. Una Trama es un mensaje que se intercambia entre una STA y un AP. Cada trama contiene información que se usa para activar un proceso del protocolo. La información contenida en cada trama, depende del proceso que se ejecute. Para fines de este trabajo de investigación, nos enfocaremos en los 17 sub-tipos de tramas del protocolo IEEE 802.11. Los sub-tipos de tramas que analizaremos se muestran en el tabla 1.

Tabla 1. Sub-tipos de tramas del IEEE 802.11

1	<i>Beacon</i>
2	<i>Probe Request</i>
3	<i>Probe Response</i>
4	<i>Authentication</i>
5	<i>Deauthentication</i>
6	<i>Association Request</i>
7	<i>Association Response</i>
8	<i>Disassociation</i>
9	<i>Reassociation Request</i>
10	<i>Reassociation Response</i>
11	<i>PS Request</i>
12	<i>PS Response</i>
13	<i>PS POLL</i>
14	<i>RTS</i>
15	<i>CTS</i>
16	<i>DATA</i>
17	<i>ACK</i>

La figura 1 representa la interacción entre un AP y cualquier STA en una infraestructura de red IEEE 802.11. El estado 1 en la figura 1 representa a las estaciones que no han obtenido privilegios (no se han autenticado ni se han asociado). En cada estado de la FSM sólo ciertos sub-tipos de tramas pueden ser intercambiadas. En el contexto de una red inalámbrica, sin una infraestructura física, el atacante tiene gran flexibilidad en decidir dónde y cuándo atacar.

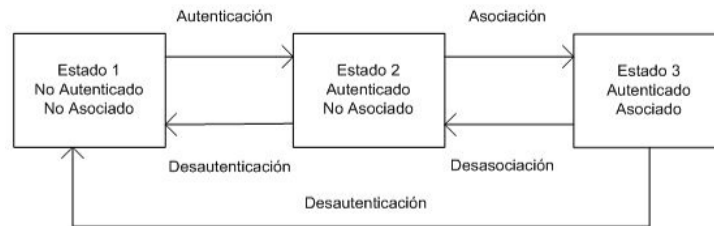


Figura 1. Máquina de Estados del IEEE 802.11

3. Problemática de seguridad del IEEE 802.11

La capa MAC del protocolo IEEE 802.11 incorpora características como la habilidad para descubrir redes, unirse y separarse de ellas, así como la coordinación para acceder al medio. Las vulnerabilidades que se presentan en el protocolo provienen de esta funcionalidad y se pueden dividir en dos categorías: de identidad y de vulnerabilidades en el control de acceso al medio.

3.1. Vulnerabilidades de identidad

Las vulnerabilidades de identidad surgen de la confianza implícita en la dirección MAC del emisor. Entonces un atacante puede engañar a otras estaciones y solicitar servicios de capa MAC, haciéndose pasar por un usuario válido. Por ejemplo, en un ataque de desautenticación, un adversario puede pretender ser un AP o un cliente y enviar tramas de desautenticación, y solicitar explícitamente la desautenticación uno del otro, en respuesta, el AP o el cliente saldrán del estado autenticado y rechazarán las tramas recibidas hasta que el estado de autenticación sea reestablecido. El proceso de ahorro de energía del IEEE 802.11 también presenta vulnerabilidades de identidad. Para ahorrar energía las estaciones pueden entrar en un modo de ahorro de energía o un modo dormido. Antes de entrar en este modo, la STA anuncia su intención al AP, para que el AP guarde en *buffer* las tramas de datos dirigidas a la STA y, cuando la STA despierte, recupere esas tramas. La presencia de paquetes en *buffer* es anunciada mediante paquetes difundidos periódicamente por el AP. Estos paquetes viajan en tramas de control que se denominan mapa de tráfico de información o TIM, por sus siglas en inglés *Traffic Information Map*. Si el mensaje TIM es falsificado por el atacante, entonces puede convencer a un cliente legítimo de que no hay tramas de datos pendientes para él y el cliente regresará al modo de ahorro de energía. Finalmente, el mecanismo de ahorro de energía recae en la sincronización entre el AP y la STA, esto es, que la STA sabe cuándo despertar. Mediante la falsificación de estas tramas, un atacante puede causar que un cliente salga de sincronía con el AP y fallar en la recuperación de las tramas pendientes.

3.2. Vulnerabilidades de acceso al medio

Las vulnerabilidades de acceso al medio surgen del mecanismo de detección de portadora virtual usado para mitigar colisiones de estaciones móviles ocultas. Cada trama de control de sub-tipo petición de envío y de sub-tipo listo para enviar o RTS CTS, respectivamente, por sus siglas en inglés *Request to Send*, *Clear to Send*, del IEEE 802.11 tienen un campo denominado duración que indica el tiempo en microsegundos que el canal es reservado. Las tramas RTS y CTS se usan para sincronizar el acceso al canal cuando una terminal oculta interfiere con las transmisiones [1]. El valor del campo duración es usado para programar el vector de asignación de red o NAV, por sus siglas en inglés *Network Allocation Vector* en cada STA. Sólo cuando el NAV de una STA alcanza el valor de cero, se le es permitido transmitir. El valor máximo de un NAV es de 32767 o cercano a los 32 milisegundos. Un atacante sólo requiere transmitir alrededor de 30 tramas por segundo para saturar el canal y evitar el acceso a la red.

4. Trabajo relacionado

En esta sección describiremos las heurísticas que se han propuesto para la construcción de sistemas de detección de intrusiones en redes inalámbricas.

Fanglu Go y Tzi-Cker Chiueh, en 2005 [5], propusieron un algoritmo para detectar robo de identidad mediante desviaciones en los números de secuencia de las tramas. Lo que proponen es medir la diferencia entre números de secuencia consecutivos entre la trama i y la trama $i-1$. La mayoría de las desviaciones fueron: 0, 1, ó 2. El 88% de las desviaciones que ellos midieron son 1, el 3.3% son 0, el 5.3% son 2 y el 2.6% son mayores a 2 debido a retransmisiones o tramas perdidas. En las pruebas realizadas, ninguna de las siete interfaces probadas transmitieron tramas fuera de orden, por lo tanto, no generaron falsos positivos. La tasa de falsos negativos fue del 0.03%. A. Martínez en 2008 [9], propuso un método para detectar tramas de administración falsificadas en el protocolo IEEE 802.11. El método se basa en monitorear el tiempo entre llegadas de las tramas de sub-tipo *Beacon*. Si el tiempo entre llegadas estaba por debajo de un umbral, las tramas se consideraron maliciosas (falsificadas). Con este método lograron una tasa del 5% de falsos positivos y 0% de falsos negativos, en la detección de tramas de sub-tipo *Beacon* falsas. Yong Sheng, en 2008 [11], propuso un modelo Gaussiano mixto para describir el comportamiento de la fuerza de la señal recibida o RSS, por sus siglas en inglés *Received Signal Strength* en los dispositivos. El método se basa en que la señal recibida es difícil de falsificar. El RSS representa la potencia de transmisión menos la atenuación de la señal. La idea básica es que un dispositivo no cambia la potencia de transmisión, entonces un cambio drástico del RSS de las tramas recibidas con la misma dirección MAC fuente, sugiere un posible ataque de robo de identidad. Usan la construcción de perfiles para detectar ataques de tipo de robo de identidad. A una tasa del 3% de falsos positivos, detectaron el 73.4%, 89.6% y hasta un 97.8% de los ataques. Desventajas: los valores medidos del RSS dependen de la distancia entre emisor y receptor, por lo que, el atacante y la víctima deben estar separados una distancia mayor a 3

metros. Además asumieron que la estación válida se mantenía estática. En consecuencia, el método sólo es efectivo si los dispositivos no presentan movilidad. Qing Li en 2007 [8], propuso usar relaciones monotónicas en los números de secuencia de las tramas inalámbrica y complementan el método propuesto con un análisis de la distribución del tiempo entre llegadas de las tramas que recibe un AP. Usan la prueba de ji-cuadrada para comprobar la distribución de probabilidad de tiempos entre llegadas de las tramas entrantes a un AP bajo la suposición de que el tiempo entre llegadas de una fuente disminuye considerablemente si dos fuentes transmiten con la misma identidad. Los trabajos realizados por [9] y [11] presentaron una relación entre la cantidad de falsos positivos y la ubicación geográfica. Al cambiar la distancia física entre dispositivos, la detección se vuelve complicada. Creemos que un análisis en la coherencia de los mensajes intercambiados es una visión más general y, permitirá describir las interacciones en redes dinámicas.

5. Metodología

Como solución al problema de detección de ataques en redes inalámbricas, proponemos una metodología que se presenta a continuación. La metodología está conformada por bloques de construcción, los cuales requieren la aplicación de métodos para el análisis de secuencias y clasificación de datos. La metodología de detección se sustenta en el análisis de bitácoras de tráfico del protocolo IEEE 802.11. Las bitácoras son producidas por la ejecución del protocolo al comunicarse una STA con un AP. Al conjunto de tramas intercambiadas entre una STA y un AP lo denominaremos conversación. Una conversación es una sucesión de tramas, esta sucesión refleja los estados que transita la interacción STA-AP en un periodo de tiempo determinado.

5.1. Separador de sesiones

Las conversaciones se tomarán aleatoriamente mediante un selector en diferentes horas del día y en distintas fechas, dentro de periodos de tiempo de una hora. Una vez separadas las conversaciones realizaremos una traducción para representar una trama de capa 2 como un símbolo, de esta manera, una conversación la representaremos como una sucesión de símbolos, donde cada símbolo representa una trama de capa 2. Primero describiremos la metodología propuesta para representar tramas de tráfico de capa 2 por medio de símbolos. La estructura del separador se presenta en la figura 2. En los diagramas nosotros representamos a los procesos como círculos y a un conjunto de datos como rectángulos. El separador de conversaciones recibirá un conjunto de tramas contenidas en una ventana W_i de tamaño T . Por cada conversación se generará un clasificador que: cada vez que reciba un número t de tramas, correspondiente a las intercambiadas entre una STA y un AP producirá una secuencia de símbolos que será analizada para obtener una caracterización de conversaciones en términos de símbolos que

pertenecen al alfabeto Σ .

$$\Sigma = \{s_1, s_2, \dots, s_m\}$$

donde m es la cardinalidad del alfabeto.

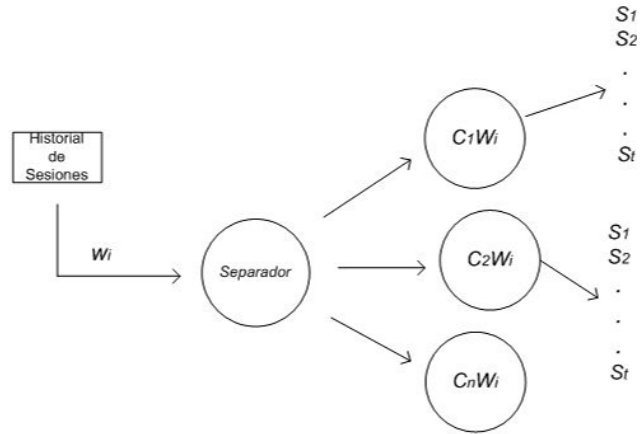


Figura 2. Arquitecturatura del Separador

5.2. Aglomerador de Sesiones

El siguiente bloque de construcción se presenta en la figura 3.

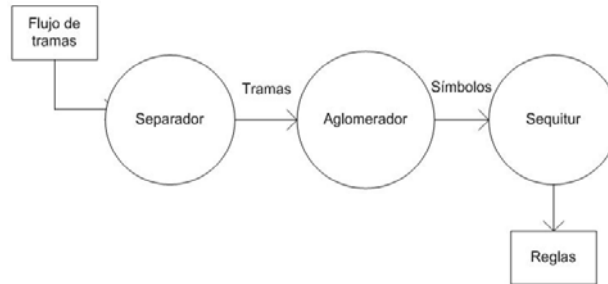


Figura 3. Obtención de Reglas

Los pasos que seguimos se describen a continuación. Para representar una trama de capa 2 como un símbolo, proponemos hacer una abstracción tomando en cuenta elementos mínimos, que describan correctamente una trama. Una trama típica de capa 2, tiene los siguientes campos de información:

1. Información de la trama inalámbrica.
2. Información del IEEE 802.11 (con cerca de 20 sub-campos, como la dirección MAC fuente, el identificador de red, etc.)
3. Datos cifrados (con más sub-campos.)
4. Datos en crudo.

Consideramos que no todos los campos son necesarios para caracterizar tráfico, es por ello que nos enfocaremos sólo en los campos que intervienen directamente en el cambio de estado de la FSM [7]. Este enfoque es importante porque el objetivo de los ataques DoS es precisamente, producir un cambio de estado a nombre de uno de los participantes [3]. Para ello, el atacante se basa en la construcción de tramas de capa 2, [6] con los valores de los campos modificados. Por otra parte, cuando un atacante genera tramas falsas a nombre de un participante, y logra un cambio de estado, la sucesión de tramas presentará incoherencias. A continuación enumeramos los campos que generan cambios en la FSM, ellos son:

1. *Subtype*: Sub-tipo de trama.
2. *Duration*: Duración de apartado de canal.
3. *SN*: Número de secuencia de trama.
4. *Fragmented*: Bandera de trama fragmentada.
5. *More Data*: Bandera de datos pendientes.
6. *Retry*: Bandera de reintento de envío de trama.
7. *ToDs*: Bandera que indica que la trama va dirigida hacia la red.
8. *Power Management*: Bandera que indica modo ahorro de energía.

Al analizar las tramas emitidas por el AP y una STA, toma importancia el campo: SN, ya que en condiciones normales se comporta de manera lineal en incrementos de 1 en un rango de valores de 0 a 4096, una vez que el valor llega a 4096 se reinicia desde 0. Las desviaciones en el patrón son mínimas. Para establecer el rango de variación calcularemos la diferencia entre el número de secuencia de la trama i y el número de secuencia de la trama $i-1$.

$$\Delta_n = SN_i - SN_{i-1}$$

Una vez obtenida la matriz con los campos y las diferencias en los números de secuencia entre tramas consecutivas, aplicamos mapas auto organizables o SOM, por sus siglas inglés *Self Organized Maps* para identificar patrones y agrupamientos de los campos contenidos en las tramas. De la matriz de resultados de SOM tomamos un número de grupos K , este número será el número de grupos que alimentamos al algoritmo K-medias con el objetivo de comprobar que los resultados arrojados por SOM son útiles para generar grupos de campos que representen una trama. Con ello agrupamos los valores de los campos para hacer un mejor razonamiento del intercambio de tramas, en lugar de usar las posibles combinaciones de campos. Una vez obtenidos los grupos de cada sub-tipo de trama inferimos reglas por medio del algoritmo C4.5 con el objetivo clasificar una trama mediante los valores de los campos contenidos en los campos

2 al 8 arriba definidos. De esta manera cada regla definió un símbolo, es decir, por cada trama clasificada correctamente de acuerdo a los valores de sus parámetros se generará un símbolo, además de un símbolo por cada grupo de tramas que el clasificador no haya clasificado correctamente. Con este conjunto de reglas construimos el alfabeto que represente cada sub-tipo de trama en condiciones de tráfico normal. Cada sub-tipo de trama estará definido por un conjunto símbolos. Con esta metodología obtuvimos la cardinalidad del alfabeto y una representación de cada trama sensible a variaciones en los valores de sus campos.

Tabla 2. Traducción de Tramas a Símbolos

Tramas de capa 2							Símbolo
Tiempo	MAC Fuente	MAC Destino	Campos				
0.365261	00:17:df:7d:a1:e0	ff:ff:ff:ff:ff:ff	0x08	0	2563	0 0 0 0 0	8001
2.314951	00:17:df:7d:a1:e0	ff:ff:ff:ff:ff:ff	0x08	0	2582	19 0 0 0 0	8001
3.082924	00:17:df:7d:a1:e0	ff:ff:ff:ff:ff:ff	0x08	0	2589	7 0 0 0 0	8001
4.277253	00:17:df:7d:a1:e0	ff:ff:ff:ff:ff:ff	0x08	0	2612	23 0 0 0 0	8004
47.054366	00:21:6b:27:d7:4c	00:21:7c:98:6c:09	0x20	44	1297	0 0 0 1 0	20001
47.060497	00:21:7c:98:6c:09	00:21:6b:27:d7:4c	0x20	44	3819	0 0 0 0 0	20002
47.067129	00:21:6b:27:d7:4c	00:21:7c:98:6c:09	0x20	44	1298	0 0 0 1 0	20001
47.073764	00:21:7c:98:6c:09	00:21:6b:27:d7:4c	0x20	44	3820	0 0 0 0 0	20002

En la tabla 2 mostramos un ejemplo de la traducción de tramas a símbolos. En este ejemplo se mapean cuatro tramas de sub-tipo *Beacon* (0x08) y cuatro tramas de sub-tipo *Data* (0x20) con sus parámetros, a símbolos S_i . De acuerdo a las reglas generadas por C4.5, podemos mapear una trama de distintos sub-tipos a símbolos en función de los parámetros de los campos que modifican la FSM. De esta manera obtuvimos 5 símbolos para representar una trama de sub-tipo *Beacon*. Las reglas generadas por C4.5 son:

1. Si el campo $Duration > 6060 - > S_1$
2. Si los campos $(Duration \leq 6060) \mathcal{E} (\Delta_n > 1027) - > S_2$
3. Si los campos $(Duration \leq 6060) \mathcal{E} (\Delta_n \leq 1027) - > S_3$
4. Si los campos $(Duration \leq 6060) \mathcal{E} (22 \geq \Delta_n \leq 1027) - > S_4$

Por otra parte, las tramas de sub-tipo *Beacon* que contengan valores fuera de estos rangos serán clasificadas como S_5 . Otro ejemplo de este mapeo son las cuatro tramas de sub-tipo *Data*, las reglas que obtuvimos se enumeran a continuación:

1. Si el campo $(Moredata \leq 0 \mathcal{E} tods > 0) - > s_6$
2. Si el campo $(Moredata \leq 0 \mathcal{E} tods \leq 0) \mathcal{E} duration > 31232 - > s_7$
3. Si el campo $(Moredata \leq 0 \mathcal{E} tods \leq 0) \mathcal{E} duration \leq 31232 - > s_8$
4. Si el campo $(Moredata moredata > 0) - > s_9$

Las tramas de sub-tipo *Data* que contengan valores fuera de estos rangos serán clasificadas como S_{10} . Es así que al aplicar esta metodología obtuvimos un alfabeto Σ con cardinalidad $m = 77$.

Para la construcción del modelo de normalidad usamos el algoritmo *Sequitur*[4], con el objetivo de detectar estructuras gramaticales repetitivas, que proporcionan poca información, y que por lo tanto pueden comprimirse y descartarse en un intento por detectar una intrusión. Nosotros redujimos sesiones identificando secuencias de tramas de capa 2 de ocurrencia frecuente, después, las reemplazamos, cada una por un nuevo meta-símbolo. Cada repetición da lugar a una regla de producción en la gramática y la sub-secuencia repetida es reemplazada por un símbolo no terminal, denominado meta-símbolo. Con esta finalidad aplicamos *Sequitur*, que es un método que infiere jerarquías de composición en la estructura de una cadena de entrada [4]. *Sequitur* detecta repetición en una cadena de entrada, y la factoriza mediante la formación de reglas en una gramática. Cuando forma reglas gramaticales, genera reglas en las que se comparten uno o más digramas, o aquellos que se utilizan sólo una vez en la producción de la cadena de entrada. Estas dos limitaciones permiten a *Sequitur* producir las reglas gramaticales más cortas que puedan generar la cadena de entrada.

Formalmente, especificamos la gramática como sigue. Sea Σ el conjunto de símbolos terminales, representando tramas; sea $N = \{n_1, \dots, n_k\}$, el conjunto de no terminales, conteniendo meta-símbolos; y, finalmente, sea δ el símbolo de inicio, entonces $\delta \notin N \cup \Sigma$. Cada regla de producción es de la forma: $n_k \rightarrow x_1 \dots x_n$, donde $n_k \in N \cup S$ es la regla más hacia la izquierda, y donde $x_1 \dots x_n$, con $x_i \in N \cup \Sigma$, es la regla más a la derecha de N . δ , entonces, es la regla más hacia la izquierda de la regla de producción. El algoritmo recibe una sesión B_i y trabaja secuencialmente leyendo los símbolos contenidos en ella en busca de sub-secuencias que se repiten dos o más veces, denotadas por $\langle sc_1 sc_2 \dots sc_l \rangle$. A cada sub-secuencia se le asigna un meta-símbolo m , que la identifica y con la que es reemplazada en B_i . Debido a que este algoritmo es recursivo, un meta-símbolo puede representar no sólo una sub-secuencia de tramas, sino también de meta-símbolos o una combinación de ambos. La salida de la reducción de una bitácora es una secuencia reducida B_i^r , y un modelo de reducción denotado por $R = \{m_1, m_2, \dots, m_a\}$, tal que al remplazar recursivamente cada m_i por su correspondiente regla de producción en B_i^r , se obtiene la secuencia original B_i . Este proceso se presenta en la figura 3.

6. Construcción del compresor

Con el conjunto reglas obtenidas por *Sequitur*, construimos un compresor. Para una compresión eficiente el compresor fué alimentado con un porcentaje del total de reglas generadas por *Sequitur*. En esta etapa implementamos el algoritmo *Boyer-Moore* [2]. Este algoritmo alinea un patrón (una cadena de símbolos P) con otra cadena de símbolos Q , donde $Q > P$ y luego comprueba si P coincide con los símbolos de Q . Después de que la comparación es completada, P se desplaza a la derecha en relación a Q . Este método examina si el patrón P se

encuentra en Q empezando de derecha a izquierda, esto normalmente le permite tener un factor significativamente más bajo que otros algoritmos de búsqueda. Para una secuencia de longitud n y patrón fijo de longitud m , es $\frac{n}{m}$ en el mejor caso, solo uno en m caracteres necesitan ser comprobados. Entre más largo es el patrón P que estamos buscando, el algoritmo suele ser más rápido para encontrarlo. En función de esta idea ordenamos el conjunto de reglas generadas por *Sequitur*, el criterio a seguir fué ordenar las reglas más largas y las más frecuentes. Posteriormente tomamos un porcentaje del total de reglas obtenidas, con este sub-conjunto de reglas, se redujo cada sesión para obtener un porcentaje de reducción. Finalmente aplicamos el mismo conjunto de reglas al total de sesiones. Para la siguiente etapa de construcción del modelo, proponemos usar modelos ocultos de Markov o HMM, por sus siglas en inglés *Hidden Markov Models* [10]. El siguiente paso será comparar el modelo de una bitácora de validación HMM' contra el modelo del historial HMM canónico, figura 4, con el objetivo de encontrar posibles anomalías.

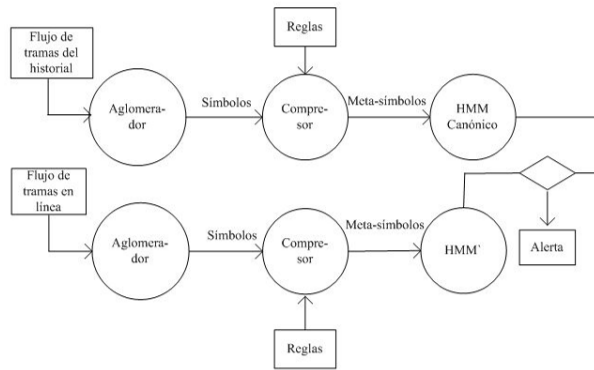


Figura 4. Arquitectura del Detector

7. Resultados

Los datos analizados son un conjunto de bitácoras que contienen una secuencia de tramas de capa 2 intercambiadas entre una STA y un AP. Seleccionamos 25 conversaciones entre diferentes parejas STA-AP del tráfico capturado en el edificio denominado *Aulas 1* del campus. Las conversaciones se tomaron aleatoriamente en diferentes horas del día y en distintas fechas, dentro de periodos de tiempo de 1 hora. Estas conversaciones representan interacciones entre STA y AP donde un usuario envía y recibe datos de la red, donde un usuario nuevo se asocia y autentica a la red del campus e interacciones ente diferentes AP's para transferir datos y mentener sincronía. El tamaño de las conversaciones fueron

de 548 tramas, la más corta y de 35259 tramas, la más larga. El total de reglas obtenidas con *Sequitur* fue de 1635. Con estos experimentos hemos logrado compactar satisfactoriamente las bitácoras, sin perder información importante. Los resultados de nuestro análisis reflejan un promedio de reducción por cada bitácora cercano al 92%, usando solamente el 20% de las reglas de mayor longitud y mayor frecuencia, esto se muestra en la tabla 3.

Tabla 3. Porcentajes de Reducción

Número de reglas	% De reglas	% De reducción
1635	100	98.2997
1380	80	97.4999
818	50	96.0227
327	20	92.0842

8. Conclusiones

Para la siguiente etapa de construcción del IDS, debemos considerar el comportamiento que presentaría un ataque, esperamos que una secuencia anómala, será difícil de comprimir, debido a patrones diferentes a los de comportamiento normal y a la persistencia del ataque. También debemos considerar que hay ataques que son efectivos con solo unas cuantas tramas, entonces, estos tipos de ataques serán detectables por la incoherencia que presentarían respecto del comportamiento normal. Es necesario considerar una arquitectura de HMM que sea capaz de diferenciar tanto secuencias que son difíciles de comprimir, como aquellas que, en longitud parecen normales pero que realmente presentan una incoherencia en el patrón con el que se haya entrenado al modelo.

Referencias

1. Bernaschi, M., Ferreri, F., Valcamonici, L.: Access points vulnerabilities to dos attacks in 802.11 networks. *Wireless Networks* 14(2), 159–169 (2008)
2. Boyer, R.S., Moore, J.S.: A fast string searching algorithm. *Commun. ACM* 20(10), 762–772 (1977)
3. Ezeife, C.I., Ejelike, M., Aggarwal, A.K.: Wids: a sensor-based online mining wireless intrusion detection system. In: Desai, B.C. (ed.) *IDEAS*. ACM International Conference Proceeding Series, vol. 299, pp. 255–261. ACM (2008)
4. Forrest, S., Hofmeyr, S.A., Somayaji, A., Longstaff, T.A.: A sense of self for unix processes. In: *IEEE Symposium on Security and Privacy*. pp. 120–128. IEEE Computer Society (1996)
5. Guo, F., cker Chiueh, T.: Sequence number-based mac address spoof detection. In: Valdes, A., Zamboni, D. (eds.) *RAID*. Lecture Notes in Computer Science, vol. 3858, pp. 309–329. Springer (2005)

6. Haghani, S., Beaulieu, N.C.: Performance of $s + n$ selection diversity receivers in correlated rician and rayleigh fading. *IEEE Transactions on Wireless Communications* 7(1), 146–154 (2008)
7. Könings, B., Schaub, F., Kargl, F., Dietzel, S.: Channel switch and quiet attack: New dos attacks exploiting the 802.11 standard. In: LCN. pp. 14–21. IEEE (2009)
8. Li, Q., Trappe, W.: Detecting spoofing and anomalous traffic in wireless networks via forge-resistant relationships. *IEEE Transactions on Information Forensics and Security* 2(4), 793–808 (2007)
9. Martínez, A., Zurutuza, U., Uribeetxeberria, R., Fernández, M., Lizarraga, J., Serna, A., Vélez, I.: Beacon frame spoofing attack detection in ieee 802.11 networks. In: ARES. pp. 520–525. IEEE Computer Society (2008)
10. Serralheiro, A.J., Ephraim, Y., Rabiner, L.R.: On nonstationary hidden markov modeling of speech signals. In: EUROSPEECH. pp. 1159–1162. ISCA (1989)
11. Sheng, Y., Tan, K., Chen, G., Kotz, D., Campbell, A.: Detecting 802.11 mac layer spoofing using received signal strength. In: INFOCOM. pp. 1768–1776. IEEE (2008)

Procesamiento de imágenes

Orden de barrido optimizado para el cancelamiento de imágenes intra dañadas

Kenia Picos-Espinoza¹, Mireya S. García-Vázquez¹ y Alejandro A. Ramírez-Acosta²

¹ IPN-CITEDI, Ave. del Parque No. 1310, Mesa de Otay, Tijuana BC, México

² MIRAL R&D, 1047, Palm Garden, Imperial Beach, 91932, USA
{kpicos,mgarciav}@citedi.mx, ramacos10@hotmail.com

Resumen La transmisión de video comprimido a través de canales propensos a errores puede resultar en un conjunto de pérdidas o errores de datos que pueden degradar significativamente la calidad de la imagen. En el caso del error en intra-cuadros, el cancelamiento de error espacial se aplica en el decodificador. En cuanto a este tipo de cancelamiento, la mayoría de la literatura se enfoca en la aplicación de la técnica de promedio ponderado, incluida en el estándar H.264 como fase no-normativa. Sin embargo, algunos trabajos de investigación consideran que el algoritmo para el orden del cancelamiento es un factor que podría incrementar el desempeño de las nuevas técnicas o alguna de referencia. En este artículo, se proponen diferentes órdenes de barrido al algoritmo de referencia del estándar H.264 para mejorar el desempeño del cancelamiento de error. Nuestros resultados muestran ganancias arriba de 0.5dB en términos de la relación señal a ruido pico (PNSR).

Palabras Clave: Cancelamiento de error, video, modificación, H.264, orden de barrido, imagen intra.

1. Introducción

Los canales de comunicación han tenido un crecimiento increíble de demanda durante las últimas décadas debido a la creación de nuevos servicios. Por ejemplo, la transmisión del video e imagen a través de redes IP (*Internet Protocol*). Con el fin de hacer buen uso del ancho de banda disponible, la transmisión de datos tiene que ser codificada (comprimida). Uno de los mayores problemas para el video comprimido es que cualquier pérdida de datos puede conducir a resultados catastróficos de decodificación (secuencias distorsionadas) lo cual no es deseable. Esta pérdida de información puede ocurrir durante la transmisión del video a través de canales propensos a errores, tales como las redes inalámbricas. Dado este problema, se han estudiado y desarrollado una serie de técnicas, durante las dos últimas décadas [1]. Estas técnicas son llamadas comúnmente como Técnicas de Resistencia al Error (*Error Resilience, ER*).

Asumiendo un sistema de comunicación como el que se muestra en la Figura 1, la ER trata de anticipar o manejar las pérdidas de datos, dependiendo en dónde las acciones se llevan a cabo en el sistema [1]. Técnicas avanzadas de cancelamiento de error se han enfocado en prevenir las pérdidas de datos, las cuales son desempeñadas por el codificador. El cancelamiento de error mediante técnicas de post-procesamiento

trata de estimar o reconstruir las pérdidas de datos, mediante métodos de interpolación u otras operaciones matemáticas. Estas técnicas son implementadas en el decodificador. Finalmente, si el codificador y el decodificador trabajan juntos para prevenir y/o reconstruir las pérdidas de datos, a estas técnicas se les conoce como cancelamiento de error interactivo o híbrido.

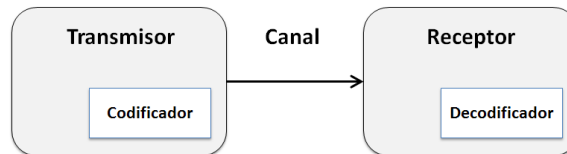


Fig. 1. Sistema básico de Comunicación.

El cancelamiento de error (CE) mediante Post-Procesamiento se encuentra dividido en tres categorías principales: Cancelamiento de error espacial (CEE), Cancelamiento de error temporal (CET) y Cancelamiento de error híbrido (CEH). En este artículo, nos hemos enfocado en CEE. Por lo tanto, el propósito de este artículo es presentar las diferentes optimizaciones que realizamos al algoritmo del Orden de barrido (OB, Scanning) para incrementar el desempeño de la técnica de cancelamiento de error espacial para imágenes Intra en el estándar H.264. Esto se realiza con el objetivo de mejorar el desempeño de la técnica de referencia de promedio ponderado (PP) en términos de PSNR. Los resultados experimentales muestran ganancias arriba de 0.5 dB. De esta manera se puede ver que es fundamental realizar un buen orden de barrido para tener un buen desempeño con la técnica de Cancelamiento de error.

Este artículo se encuentra estructurado como sigue: en la sección 2, se presenta una breve reseña del cancelamiento de error espacial en el marco del codificador H.264. En la sección 3 se describen las diferentes optimizaciones del OB para CEE. Los resultados son presentados en la sección 4. Por último, en la sección 5 se plasman las conclusiones.

2. Cancelamiento de error espacial

Las técnicas de CEE comúnmente son utilizadas por imágenes codificadas Intra (imágenes independientemente codificadas sin referencia) en el contexto de la norma MPEG-4 H.264/AVC [2]. Como su nombre lo indica, estas técnicas son operadas pixel por pixel (en el dominio espacial). Esto significa que todos y cada uno de los pixeles pertenecientes al área de pérdida son reconstruidos uno a la vez. Para lograr la reconstrucción, se toman en cuenta algunas consideraciones espaciales. Un ejemplo de esto, es que en muchas de las escenas encontradas en la naturaleza predominan las bajas frecuencias; las cuales se presentan en las imágenes como cambios suaves en las intensidades de los pixeles. Sin embargo, esto es sólo una manera de lograr CE, de muchas más que existen.

Viktor Varsa et al. [3] propuso a la ITU (Unión Internacional de Telecomunicaciones) adoptar algoritmos de cancelamiento de error no-normativo. Una de sus metodologías propuestas es para imágenes codificadas Intra. En este algoritmo, cada pixel es cancelado haciendo uso de los cuatro pixeles vecinos más cercanos que no hayan sido

reconstruidos. La Figura 2 es un ejemplo de esta clase de CEE llamada promedio ponderado (PP).

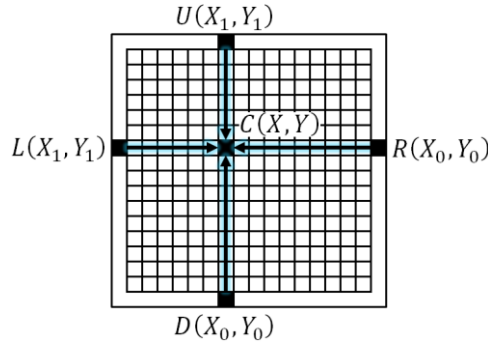


Fig. 2. Técnica de promedio ponderado.

El área en cuadrícula de la Figura 2 representa el área dañada dentro de un cuadro. Los píxeles que se encuentran ubicados en el límite exterior de esta área se consideran candidatos para el proceso de cancelamiento de píxeles. El promedio ponderado de píxeles se calcula de la siguiente manera:

- En primer lugar, para todos y cada uno de los píxeles, se requiere el cálculo del incremento en las direcciones horizontal y vertical; esto se realiza utilizando los píxeles más cercanos a la derecha, izquierda, arriba y abajo de los límites del borde del área.

$$\Delta x = \frac{R(X_0, Y_0) - L(X_1, Y_1)}{X_1 - X_0} \quad (1)$$

Δx : Incremento en dirección horizontal
 Δy : Incremento en dirección vertical
 $R(X_0, Y_0)$: píxel localizado a la derecha
 $L(X_1, Y_1)$: píxel localizado a la izquierda
 $D(X_0, Y_0)$: píxel localizado abajo
 $U(X_1, Y_1)$: píxel localizado arriba

- Y finalmente se calcula promediando:

$$C(X, Y) = \frac{[X_0 + (X_1 - X_0)\Delta x] + [Y_0 + (Y_1 - Y_0)\Delta y]}{2} \quad (2)$$

donde $C(X, Y)$ es un píxel bajo reconstrucción. Es importante destacar que este tipo de cancelamiento da buenos resultados cuando se trata de áreas *suaves*. Sin embargo, este no es el único esquema de reconstrucción espacial, existen muchos más que trabajan para otro tipo de áreas dañadas. Dado que la definición de bordes juega un papel importante en la percepción visual humana, otra consideración importante es la existencia de bordes dentro del área de pérdidas. El cancelamiento de error basado en interpolación direccional es propuesto en [4]. Este esquema convoluciona valores de los píxeles de los bloques de la imagen alrededor del área de pérdidas con máscaras direccionales predeterminadas, tratando de encontrar si los bordes predominantes

existen en la región vecina. Las máscaras de Sobel se emplean frecuentemente en este tipo de situaciones [5].

3. Orden de cancelamiento de error espacial en intra-cuadro

Como se ha mencionado en la sección anterior, la transmisión de video comprimido a través de canales propensos a error puede resultar en pérdidas o errores, lo cual pueden degradar la calidad de la imagen significativamente. En el caso de error en intra-cuadro, el CEE se aplica en el lado del decodificador. La mayoría de la literatura [4-8] se enfoca en superar el rendimiento de la técnica de referencia del promedio ponderado incluida en la norma como fase no-normativa (explicada en la sección 2). Sin embargo, pocos de los esfuerzos en esta área de investigación consideran el papel que juega el algoritmo del orden de cancelamiento como un factor que puede incrementar el desempeño de los nuevos algoritmos y los de referencia. Como se sabe, la eficacia del cancelamiento de error de un macrobloque (MB) de dimensiones 16x16 píxeles depende altamente de sus condiciones de vecindad [2], por lo tanto, en este artículo, se explora la posibilidad de modificar el orden de barrido (OB) del proceso, con el propósito de determinar si el orden de cancelamiento aplicado al algoritmo de CEE de referencia puede modificar su desempeño en Intra-cuadros.

3.1. Orden de barrido en el algoritmo de referencia

El OB es el orden en el cual los MB con pérdida o dañados son buscados y cancelados en el Intra-cuadro. Para el cumplimiento del CE en el software de referencia [9], existen dos OB dependiendo de qué clase de cuadro ha sido decodificado (Inter o Intra). El OB para Intra-cuadros (Figura 3) inicia en el MB de la parte más izquierda superior (el cual pertenece a la primera columna) y va hacia abajo hasta el último MB (de la misma columna), luego la segunda columna (de izquierda a derecha) se analiza y así hasta el último MB de la última columna (inferior derecha) [3].

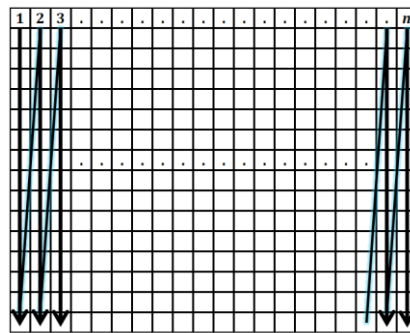


Fig. 3. Orden de barrido para Intra-cuadro.

Por otro lado, el OB para Inter-cuadros, mostrado en la Figura 4, inicia buscando en el MB de la parte superior izquierda hasta alcanzar el último MB de la columna actual, y así se mueve hasta la última columna (primera vista de derecha a izquierda).

Los números en el primer renglón (Figura 4) denotan la secuencia en el cual se realiza el barrido. El proceso converge en las columnas de en medio.

3.2. Modificaciones de orden de barrido en el algoritmo de referencia

Una cuestión importante es que dada una pérdida de MB, el OB del algoritmo de referencia no considera los MB previamente cancelados para el CE. En otras palabras, si durante la tarea de cancelamiento de un MB específico, se ha encontrado que al menos existen 2 MBs vecinos correctos (no cancelados previamente), el cancelamiento se desempeñará sin la necesidad de buscar el MB vecino cancelado previamente.

Por otro lado, si el algoritmo no puede encontrar al menos 2 MBs vecinos correctos, (no cancelados previamente), entonces los MBs previamente cancelados se tomarán en cuenta en la tarea de cancelamiento. La Figura 5 muestra el mapa de estado de los MBs en el decodificador, el cual describe situaciones para todos y cada uno de los MBs en un cuadro del video. El estado 3 denota un MB recibido correctamente, el estado 2 representa el MB cancelado y el estado 0 indica pérdida del MB.

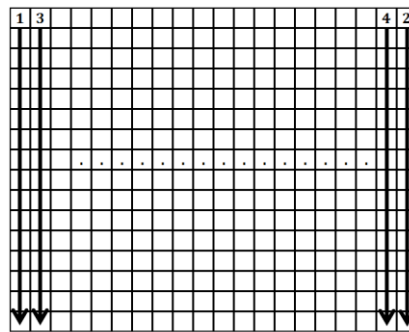


Fig. 4. Orden de barrido para Inter-cuadro.

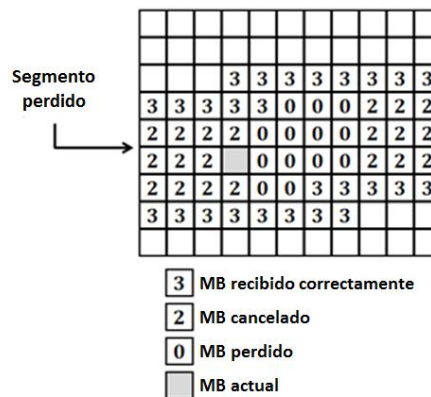


Fig. 5. Mapa de estado de los MBs.

El OB para el CE no considera el error de propagación, el cual es el tópico principal para CE. Considerando esto, se hacen modificaciones al algoritmo del OB para

CEE en el software de referencia del estándar MPEG-4 H.264/AVC [9]. La Tabla 1 muestra las modificaciones y sus descripciones respectivas:

Tabla 1. Modificaciones de OB en el algoritmo de referencia del CEE.

Modificación	Descripción
Referencia	Esta no es una modificación, sino es el OB dado en el codificador H.264 para CEE (Intra-cuadros)
α	En vez de buscar el primer MB vecino no reconstruido (que es el caso del OB de referencia), esta modificación incluye todos los MBs adyacentes disponibles (aún los que han sido cancelados previamente).
β	Como se describe en la sección 3 (el OB en el algoritmo de referencia de CE) los Inter e Intra cuadros tienen su propio OB para el CE; esta adaptación propone desempeñar Inter OB en Intra-cuadros.
$\alpha + \beta$	Esta modificación es una combinación de adaptaciones de α y β ; esta versión desempeña Inter OB además incluye todos los MBs vecinos disponibles (aún los que han sido cancelados previamente) desde el inicio del proceso.
γ	La propagación del error es una tarea importante cuando se trata de CE de pérdidas de MB, los errores se deberían propagar uniformemente a lo largo del cuadro, pero no ocurre así; estas modificaciones inician con el cancelamiento de pérdidas de MB que tengan cuatro MBs vecinos disponibles no cancelados previamente, continúa el proceso para aquellos con tres MBs vecinos disponibles no cancelados previamente, luego trabaja con dos MBs no cancelados previamente, y finalmente cuando tenga que trabajar con un MB vecino, se utiliza el cancelado anteriormente.
$\gamma + \alpha$	Esta versión de modificación es una combinación de α y γ .
δ	Una versión modificada de γ , esta variación inicia mediante el cancelamiento del MB con al menos tres MBs adyacentes no cancelados, y continúa con los MB que tiene al menos dos MBs vecinos (a partir de esto, no importa si los MBs disponibles han sido previamente cancelados).
$\delta + \alpha$	Esta versión de modificación es una combinación de δ y α .

La primera modificación de código (denotada por α) ha sido creado para considerar el MB cancelado previamente para CE, no importando cuántos MBs vecinos se encuentran disponibles (no cancelados previamente). La segunda ha sido un ajuste hecho al OB (denotada por β), con el fin de lograr Inter OB en Intra-cuadros. La tercera variación es una combinación de la primera y la segunda modificación (denotada por $\alpha + \beta$). La cuarta alteración (denotada por γ) se basa en la primera cancelación del MB dañado con cuatro vecinos disponibles de MBs no previamente cancelados, luego para cancelar el MB con tres vecinos disponibles, continuando con el MB dañado con dos vecinos disponibles y finalmente, terminando con los MBs que tengan solamente un MB adyacente. La quinta modificación (denotada por $\gamma + \alpha$) es una combinación de la cuarta y la primera adaptación. La sexta modificación es una versión modificada de γ . La séptima y última modificación (denotada por $\delta + \alpha$) es una combinación de la sexta y la primera adaptación.

4. Resultados

Las optimizaciones descritas previamente han sido probadas en el software de referencia del estándar MPEG-4 H.264/AVC versión 14.2 [9] con las siguientes secuencias de referencia QCIF (352x288 píxeles): *Foreman*, *Bus*, *Coastguard* y *Mobile*.

Bajo las mismas condiciones de tasa de pérdida de paquetes (TPP) del 5%, 10%, 15% y 20%, como se muestra en las Figuras 6 a la 9, los resultados demuestran que las optimizaciones $\gamma + \alpha$ ofrecen mejoras en PSNR (el objetivo de medida de calidad) para las secuencias *Foreman* y *Mobile* de 0.2915 dB en la mejor de las situaciones. Mientras tanto, la optimización $\delta + \alpha$ provee mejoras para las secuencias *Coastguard* y *Bus* de 0.5053 dB en el mejor de los casos. Las Figuras 10 a la 13 muestran los resultados para el componente de luminancia (Y) en el Intra-cuadro *Coastguard*, *Foreman* y *Bus* aplicando el OB de referencia del CE (H.264) y las optimizaciones del OB α , $\gamma + \alpha$ y $\delta + \alpha$ respectivamente.

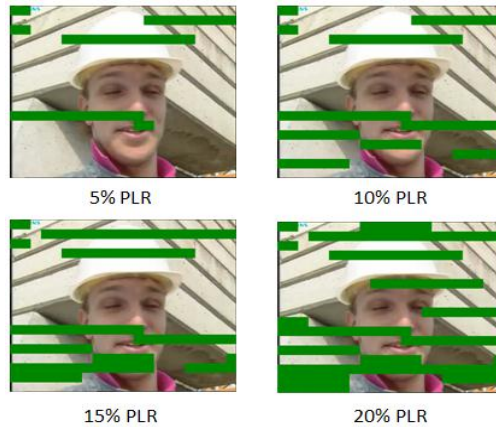


Fig. 6. Diferentes niveles de TPP para la secuencia *Foreman*.

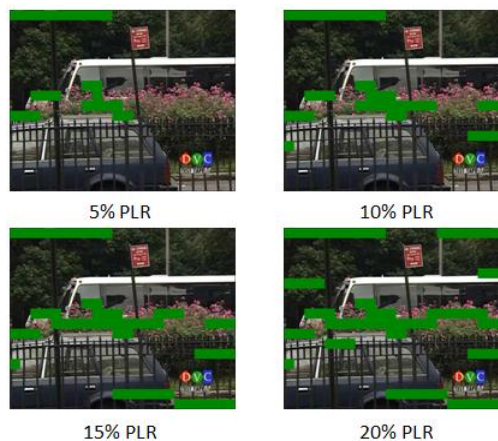


Fig. 7. Diferentes niveles de TPP para la secuencia *Bus*.

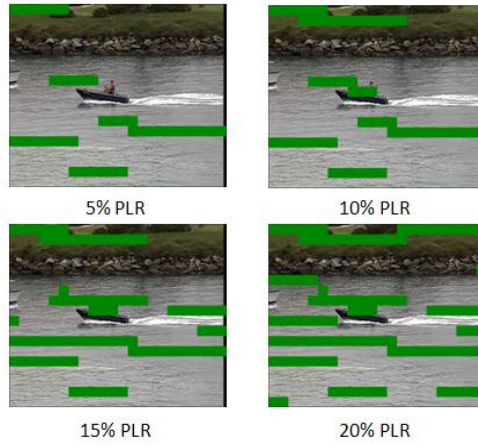


Fig. 8. Diferentes niveles de TPP para la secuencia *Coastguard*.

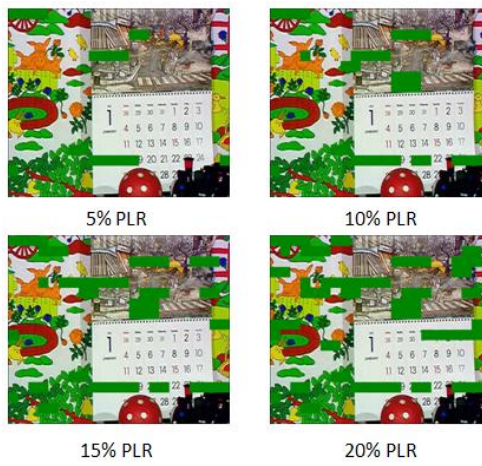


Fig. 9. Diferentes niveles de TPP para la secuencia *Mobile*.

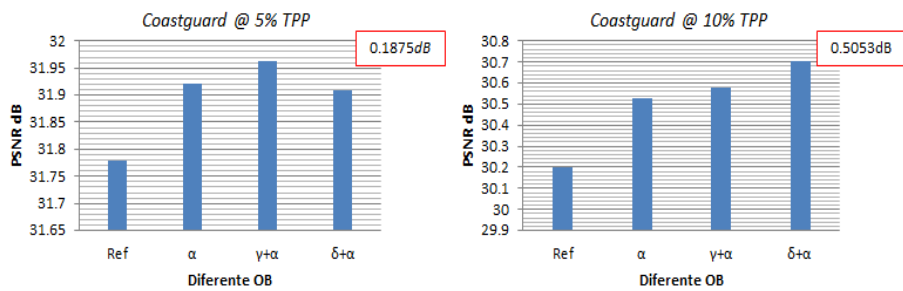


Fig. 10. *Coastguard* @ 5% y @ 10% TPP.

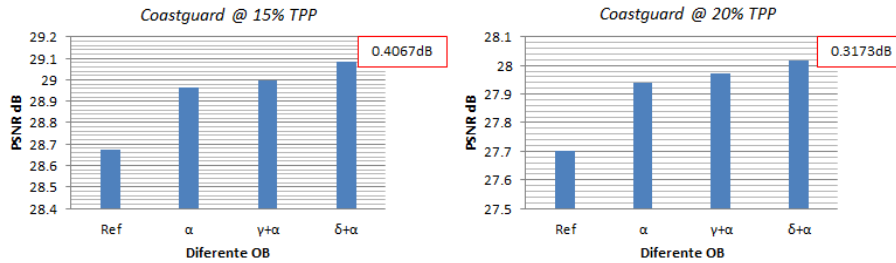


Fig. 11. Coastguard @ 15% y @ 20% TPP.

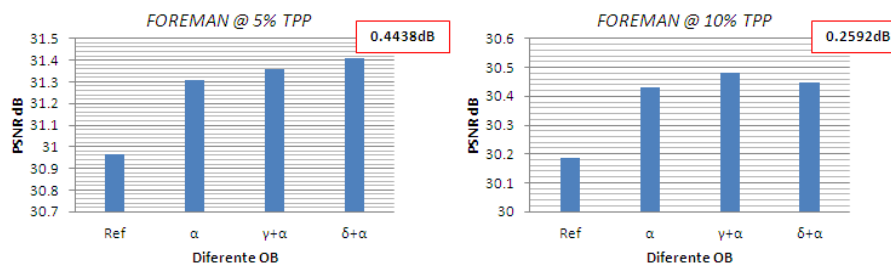


Fig. 12. Foreman @ 5% y @ 10% TPP.

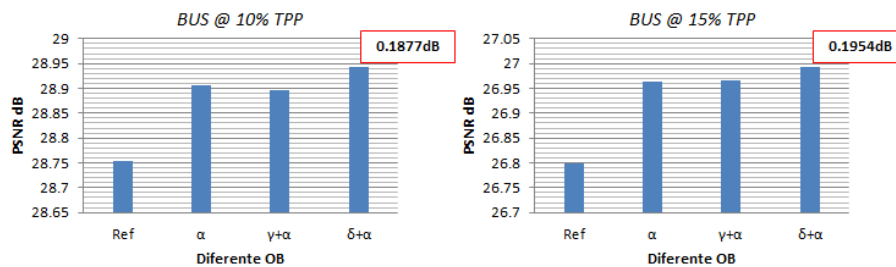


Fig. 13. Bus @ 10% y @ 15% TPP.

La versión $\delta + \alpha$ claramente supera las otras optimizaciones y brinda una mejor solución en 50% de las situaciones (para todas las secuencias). Esto puede ser atribuido a su metodología de cancelamiento en el cual los MBs previamente cancelados se incorporan dentro del proceso de cancelamiento desde el inicio (modificación α); además su propiedad de cancelamiento de MB con la mayor cantidad de MB no cancelados previamente.

5. Conclusiones

El algoritmo de cancelamiento de error juega un papel importante en el decodificador bajo una situación de pérdida de paquetes. Este es el caso de la transmisión de video bajo canales propensos a errores. Cuando los errores son presentados en Intra-

cuadros, se aplica el cancelamiento de error espacial. En este artículo se presentaron siete optimizaciones al algoritmo del orden de barrido para el cancelamiento de error espacial. Los resultados demuestran claramente la importancia de tener un algoritmo del orden de barrido apropiado para realizar un mayor desempeño en la técnica de cancelamiento de error. Las optimizaciones en el software de referencia del estándar MPEG-4 H.264/AVC versión 14.2 se llevaron a cabo de una manera satisfactoria. Los resultados experimentales muestran ganancias arriba de 0.5 dB con nuevos órdenes de barrido.

Agradecimientos. Esta investigación ha sido financiada por SIP2012 por el IPN.

Referencias

1. Wang Y., Shu Q.: Error Control and Concealment for Video Communications. Review, Proceedings of IEEE, vol. 86, pp. 974-997 (1998)
2. Richardson, I. E.: The H.264 and MPEG-4 Video Compression. John Wiley & Sons, (2003)
3. Varsa V, M. M. Hannuksela and Y. K. Wang: Non-normative error concealment algorithms, ITU-T VCEG-N62, Sept. (2001)
4. Agrafiotis D.: Spatial error concealment with edge related perceptual considerations. Signal Processing: Image Communication, ELSEVIER, pp. 130-142 (2006)
5. Suh, J., Ho, Y.: Error Concealment Based on Directional Interpolation. Transactions on Consumer Electronics, IEEE, vol. 43, pp. 295-302 (1997)
6. Petrani M.: Error Concealment of Intra Coded Video Frames. Image analysis for multimedia interactive services, IEEE Conferences, pp. 1-4 (2010)
7. Chung Y.: A Novel Intra-Frame Concealment Algorithm for H.264 AVC. International Conference on Convergence and Hybrid Information Technology, IEEE, pp. 881-886 (2008)
8. Sun Y.: A Dynamic Temporal Error Concealment Algorithm for H.264. International Conference on Multimedia Technology, IEEE, pp. 1-4 (2010)
9. H.264/AVC Codec Software: JM14.2 Video Coding Standard.

Propuesta de un algoritmo híbrido para la corrección de imágenes RAPD

Ricardo Contreras A., M. Angélica Pinninghoff J. y Alejandra Alvarado V.

Departamento de Ciencias de Computación,
Universidad de Concepción, Chile
{`rcontrer,mpinning`}@udec.cl

Resumen Este artículo describe una experiencia que combina Algoritmos Genéticos y Simulated Annealing como un mecanismo de corrección de carriles en imágenes ADN obtenidas a través de la técnica RAPD (Random Amplified Polymorphism DNA). Las imágenes RAPD son afectadas por diversos factores; entre ellos el ruido y la distorsión que impactan la calidad de las imágenes, y consecuentemente, la precisión en la interpretación de los datos. Este trabajo propone un método híbrido que emplea algoritmos genéticos, para tratar con la característica altamente combinatorial de este problema, y simulated annealing, para tratar con los óptimos locales. Los resultados obtenidos con esta aproximación sobre este problema particular muestran una mejora, respecto del uso exclusivo de algoritmos genéticos, tanto en el fitness, o calidad de los individuos, como en los tiempos de ejecución.

Palabras clave: Imágenes RAPD, cocción simulada (*simulated annealing*), algoritmos genéticos.

1. Introducción

RAPD (Randomly Amplified Polymorphism DNA) [11] es un tipo de marcador molecular que se utiliza para la verificación de identidad genética. En los últimos años, se ha utilizado RAPD para estudiar relaciones filogenéticas [1][10], mapeamiento genético [4], marcadores asociados a rasgos [9], y mapeamiento de enlaces genéticos [2]. Esta técnica ha sido usada como apoyo en muchos programas asociados a la agricultura, al sector forestal y a la reproducción de animales [5].

En la figura 1, se muestra la fotografía de una imagen RAPD. En este caso se consideraron 12 muestras, usando los carriles 1 y 14 para indicar pesos moleculares estándar. Cada uno de los carriles mostrados en la imagen está compuesto por un conjunto de trazos de diferente brillo, que corresponden a las bandas de cada carril. Se estudiaron cuatro diferentes genotipos de *Eucalyptus globulus*, incluyendo tres copias idénticas de cada genotipo (conocidos como rametos). Si los rametos son idénticos, entonces es de esperar que se obtengan patrones de bandas similares, al ser analizados en las mismas condiciones. Sin embargo, esto

no ocurre debido a, por ejemplo, errores en la rotulación de las muestras u otro tipo de deformaciones producidas durante el proceso.

La técnica RAPD consiste en la amplificación randómica de secuencias del ADN genómico usando *primers* (cebadores), los cuales comunmente tienen una longitud de 10 pb (pares base). Este proceso es llevado a cabo por la reacción en cadena de la polimerasa, conocido como PCR (polymerase chain reaction) y genera un patrón típico para una sola muestra y diferentes *primers*. Un *primer* es una cadena de ácido nucleico que sirve como punto de partida para la síntesis del ADN. Los productos PCR son separados en un gel de agarosa, lo que permite que, bajo la acción de un campo eléctrico, los fragmentos más pequeños de los productos PCR se desplacen más rápido, mientras que los más grandes lo hagan más lentamente. El gel es coloreado con una tintura (típicamente bromuro de etidio) y fotografiado para el análisis posterior de los datos. Una manera de analizar la imagen obtenida es la simple comparación visual de las diferentes bandas que se obtiene, para cada muestra. Sin embargo, puede ser un proceso tedioso cuando varias muestras con diferentes combinaciones de *primers* deben ser analizadas. Al mismo tiempo, ya que en este caso la presencia o ausencia de bandas es cuantificada, esa cuantificación puede ser subjetiva y no hay un nivel de umbral confiable, ya que las intensidades de las bandas son afectadas por varios factores (tintura, calidad del gel, reacción PCR, calidad del ADN, entre otros).

Durante el proceso de generación de una imagen RAPD, muchos factores físico-químicos afectan la electroforesis produciendo diferentes tipos de ruido, rotaciones, deformaciones y otras distorsiones anormales en la imagen. El efecto de esto es, desafortunadamente, propagado a las diferentes etapas en el análisis posterior, incluyendo visualización, extracción del background, detección de bandas y clustering, que puede conducir a conclusiones biológicas erróneas. Por esta razón, técnicas eficientes de procesamiento de imágenes pueden tener un impacto positivo sobre esas conclusiones biológicas.

Errores típicos consideran rotación de carriles, que es el problema que intentamos resolver. Este es el primer paso; una vez que los carriles son corregidos, es decir, la imagen completa muestra pendientes mínimas para cada carril, es necesario trabajar en la corrección de las bandas, un problema difícil debido a la naturaleza de las distorsiones, diferente a la distorsión de carriles. Este segundo paso, la corrección de las bandas, puede ser atacado de manera análoga; sin embargo, va más allá del alcance de este artículo.

La base para este trabajo es la experiencia descrita en [7], donde se usan algoritmos genéticos para tratar con una población de soluciones potenciales, donde las soluciones se entienden como los mejores *templates* (esquemas). Un *template* es un conjunto de líneas que representan carriles. Así, el mejor *template* es uno en el cual las líneas presentan pendientes que se aproximan de manera más cercana a las pendientes que aparecen en la imagen original. Este trabajo presentó buenas soluciones, aunque los tiempos de ejecución eran superiores a

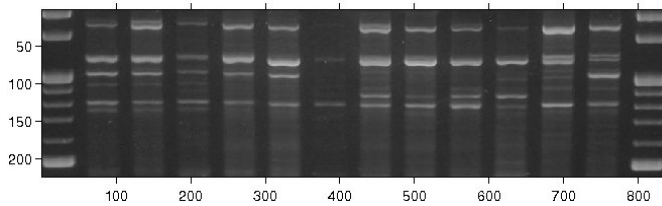


Figura 1. Una muestra de imagen RAPD con dos carriles de referencia, y 12 carriles que representan a cuatro rametos.

los esperados. El otro problema que este enfoque presenta, es la presencia de mínimos locales.

El objetivo de este trabajo es corregir las distorsiones presentes en los carriles, usando algoritmos genéticos hibridados con *Simulated Annealing* (Recocido simulado). Esto permite la comparación de dos estrategias: la primera, que considera solamente algoritmos genéticos, y la segunda, que usa algoritmos genéticos y simulated annealing como mecanismos colaborativos.

Este artículo está estructurado de la siguiente forma; la primera sección consta de la presente introducción; la segunda sección describe el problema específico a ser enfrentado; la tercera sección está dedicada a consideraciones sobre algoritmos genéticos y simulated annealing, mientras la cuarta sección muestra los resultados obtenidos con nuestra propuesta. La sección final presenta las conclusiones del trabajo.

2. El enfoque propuesto

El problema enfrentado en este trabajo puede ser formalmente establecido de la siguiente forma.

Considérese una imagen (una matriz) $A = \{a_{ij}\}, i = 1, \dots, n$ y $j = 1, \dots, m$, donde $a_{ij} \in Z^+$, y A es una imagen RAPD. Usualmente, a_{ij} está en el rango $[0, 255]$ en una imagen en escala de grises, y se utiliza a_{ij} para denotar a un elemento $A(x, y)$, donde x e y son las coordenadas del pixel.

Para tratar las distorsiones de carriles, se usa un conjunto de *templates*. Estos *templates* son líneas creadas en forma random que presentan diferentes grados de distorsión, y que están en correspondencia uno-a-uno con los carriles en la imagen RAPD original. Un buen *template*, al igual que lo considerado en [7], es aquel que refleja de manera más precisa las distorsiones que tiene la imagen RAPD bajo consideración.

El *template* es representado por una matriz L (los carriles) donde $L = \{l_{ij}\}, i = 1, \dots, n$ y $j = 1, \dots, m; l_{ij} = 0$ o $l_{ij} = 1$ (una imagen binaria), con 1 indicando que l_{ij} pertenece a una línea y 0 en caso contrario. Un procedimiento descrito en [8] se usa para detectar en forma aproximada la posición inicial de los carriles. De esta forma la generación de la matriz L se limita a aquellas regiones que corresponden a carriles en la matriz A . Debido a la rotación de los carriles, es

necesario considerar diferentes configuraciones alternativas. Si se está tratando con una imagen de 12 carriles, y si por cada línea se consideran 14 rotaciones posibles, entonces se está hablando de 12^{14} configuraciones diferentes a evaluar. Esto provoca una explosión combinatoria, que es lo que justifica el uso de una técnica como los algoritmos genéticos.

Los algoritmos genéticos permiten manipular un gran número de *templates*, escogiendo aquellos que son similares a la imagen original. Así, es necesario encontrar una función objetivo que refleje esta similitud de una manera precisa. Esta función es usada como una medida de la calidad del *template* seleccionado.

Cuando se aplica el procedimiento de corrección de carriles, los *templates* contienen líneas rectas. Diferentes *templates* mostrarán diferentes pendientes para cada línea, como se muestra en la figura 2. Un *template* contiene líneas verticales que no se intersectan, pero que no son necesariamente paralelas (en el marco restringido a la imagen).

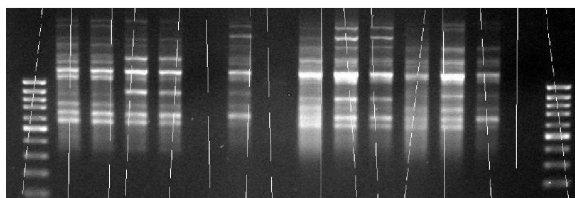


Figura 2. Un *template* de muestra para la corrección de carriles.

Los resultados obtenidos en trabajos previos eran prometedores, pero no lo suficientemente buenos como para ser aceptados como una alternativa. Con esta consideración en mente, se decidió hibridar la estrategia solución incorporando la técnica de simulated annealing. Simulated annealing es un procedimiento para optimizar una función en base a perturbaciones randómicas de una potencial solución y a una decisión probabilística sobre la posibilidad de mantener la solución mutada [3].

3. Algoritmos genéticos y *simulated annealing*

Los algoritmos genéticos (AG) son una clase particular de algoritmos evolutivos, usados para encontrar soluciones óptimas, o buenas, examinando sólo una fracción reducida del espacio de soluciones posibles. Los AGs están inspirados en la teoría de la evolución de Darwin.

En lo fundamental, los AGs se diseñan para simular procesos necesarios para la evolución en sistemas naturales, específicamente aquellos que siguen los principios de supervivencia del más adecuado. De esta forma, representan una explotación inteligente de una búsqueda randómica dentro de un espacio de búsqueda definido para resolver un problema.

La estructura de un algoritmo genético consiste de un procedimiento iterativo simple sobre una población de individuos genéticamente diferentes. Los fenotipos son evaluados de acuerdo a una función de fitness predefinida; los genotipos de los mejores individuos son copiados varias veces y luego modificados por medio de los operadores genéticos; los nuevos genotipos obtenidos en este proceso son introducidos en la población, en reemplazo de los antiguos. Este procedimiento continúa hasta que una solución *suficientemente buena* es encontrada [3].

En este trabajo, los *templates* son los cromosomas, las líneas en los templates son los genes y una línea con una pendiente particular representa el valor (alelo) que tiene un gen.

Un buen fitness indica que un *template* particular (matriz L) tiene una mayor similitud con la imagen RAPD original (matriz A).

Para evaluar un *template*, las imágenes que corresponden a las matrices A y L se superponen, obteniéndose una suma de intensidades que considera los pixeles de una vecindad dentro de un rango para cada línea. En este trabajo, este rango se determina considerando el ancho de la porción más brillante del carril. El objetivo de esto es obtener mayor precisión en la función fitness. Si una línea en el *template* coincide con un carril, se obtiene un mayor valor para la suma. En contraste con lo anterior, si no existe coincidencia, el valor es menor que en el primer caso, porque se está agregando intensidades de pixeles que corresponden al fondo de la imagen (valores cercanos a cero).

Un aspecto interesante en el presente enfoque resulta del hecho que el valor obtenido para la evaluación de cada línea es almacenado como parte del *gen*. De esta forma, la suma de las intensidades se hace solamente al momento de crear una nueva línea. Como consecuencia de esto, el tiempo de ejecución se reduce de manera considerable, en comparación con experiencias previas.

Operadores genéticos: Para este trabajo se consideraron diferentes operadores genéticos. Estos operadores se describen brevemente a continuación:

- **Selección.** La selección se lleva a cabo utilizando el mecanismo de *ruleta* [3]. Esto significa que los individuos con mejores valores de fitness tendrán mayor probabilidad de ser escogidos como padres en el proceso de reproducción.
- **Cruzamiento.** El cruzamiento se usa para intercambiar material genético, permitiendo que parte de la información genética de un individuo se combine con parte de la información genética de un individuo diferente. Por ejemplo, si se tiene dos *templates*, cada uno de ellos con $r + s$ líneas, después del cruzamiento los hijos generados resultan en: hijo 1 tendrá las primeras r líneas provenientes del *template* 1, mientras que las siguientes s líneas provendrán del *template* 2. Para el hijo 2, el proceso es levemente diferente, en cuanto a que el orden de los *templates* considerados se modifica.
- **Mutación.** Al usar este operador genético, se introduce una pequeña variación en la población de manera que se crea nuevo material genético. En este trabajo, la mutación se realiza reemplazando en forma aleatoria, con baja probabilidad, una línea particular en un *template*.

Simulated Annealing (SA) es una técnica que se inspira en la obtención de metales de diversas formas, a partir del metal en estado líquido, a través de un proceso de enfriamiento gradual, permitiendo que el material transite desde un estado inestable, desordenado, de alta energía, a un estado estable, ordenado y de baja energía.

En SA, el material es una solución candidata. Esta solución sufre una mutación y, si su energía es menor que la energía de la solución en el estado previo, la solución mutada reemplaza a la original. Si, por otra parte, la energía es mayor, entonces se produce el reemplazo de la anterior dependiendo de una probabilidad que es proporcional a la diferencia de energías. Inicialmente, cuando la temperatura del sistema es alta, las soluciones mutadas con energía relativamente altas (bajo fitness) tienen alguna posibilidad de ser mantenidas en el espacio de soluciones. La temperatura del sistema es disminuida después de n evaluaciones, reduciendo de forma efectiva la probabilidad de mantener soluciones mutadas con valores de energía altos [3].

Para este trabajo se hace uso de una variante del algoritmo SA que permite que se produzca *reannealing* (aumento de la temperatura), ya que entrega más libertad en la búsqueda por soluciones, tanto en tiempo como en espacio. El problema de encontrar la inclinación de carriles a través de SA consiste en, dada una determinada temperatura T , buscar durante a lo más K iteraciones una cantidad S de soluciones. Cada configuración o posible solución es generada a través de una determinada función, a partir de una solución encontrada durante la iteración anterior. La solución de partida es aquella proporcionada por la salida del AG, un individuo representado por un conjunto de rectas donde cada una se ajusta en cierta medida a la inclinación de cada carril de la imagen RAPD bajo análisis. Cada vez que se crea una nueva configuración ésta es evaluada; de ser mejor que la antigua, esta última se desecha, mientras que si ocurre lo contrario, se acepta bajo cierta probabilidad, que es proporcional a la temperatura.

Un parámetro, asociado a la temperatura, da cuenta de la magnitud del movimiento a realizar, a mayor temperatura, los cambios son más drásticos; a menor temperatura, los cambios son más sutiles.

Cuando el proceso finaliza, se espera seleccionar una solución donde el conjunto de rectas asociado a la imagen representa la inclinación de los carriles de mejor manera que la entregada por la configuración de partida. A partir de una configuración x , durante la aplicación de SA se producen dos posibles movimientos, para generar soluciones candidatas que representan un aumento o una disminución de la energía.

Una solución es un *template* que representa una imagen RAPD, la solución x , que debe transformarse en una solución x' a través de una modificación del *template*. Las dos posibilidades de modificación del *template* recién mencionadas son: en primer lugar, el cambio en la pendiente de una o más líneas del *template*, es decir, un movimiento de rotación; en segundo lugar un movimiento de traslación, es decir, una línea es movida hacia la izquierda o derecha en el

template, sin alterar su pendiente. En otras palabras, si llamamos x_{inf} e x_{sup} a los valores de los puntos inferiores y superiores en una línea, respectivamente, un movimiento de rotación se consigue cambiando los valores de esos puntos a $x_{inf} - \delta$ y $x_{sup} + \delta$ (o modificando a $x_{inf} + \delta$ y $x_{sup} - \delta$). De manera análoga, el movimiento de traslación se consigue desplazando los puntos originales a $x_{inf} + \delta$ y $x_{sup} + \delta$ (cambiando el signo si el movimiento es hacia el otro lado de la imagen). La Figura 3 ilustra el movimiento de rotación y el movimiento de traslación. Los valores permitidos tanto para la rotación como para el desplazamiento son disminuidos consecuentemente con la disminución de la temperatura, para evitar cambios severos en la calidad de las soluciones. Esto pensando en que temperaturas más bajas se acercan más a óptimos globales.

La elección de cuál movimiento realizar se toma aleatoriamente, aunque es posible definir un parámetro que permita privilegiar una opción por sobre la otra. En la rotación y el desplazamiento de las líneas se considera un valor máximo para δ de 10 píxeles.

El criterio de parada se definió en función de un cierto número de iteraciones. Esta decisión se debió a que la elección de una temperatura (mínima) como argumento, se podía distorsionar debido a eventuales aumentos de temperatura, realizados durante el proceso (reannealing, como se mencionó más arriba).

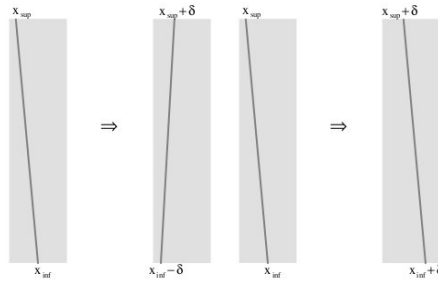


Figura 3. Esquema para rotación y desplazamiento de una línea.

4. Pruebas y resultados

Para verificar la hipótesis original, se construye un prototipo basado en AGs, hibridado con SA, obteniéndose mejoras en fitness y en tiempo de ejecución con respecto a trabajos previos, como el presentado en [6].

El problema de la detección de la inclinación de carriles se aborda haciendo uso de un algoritmo híbrido que combina ambos métodos mencionados. Se plantean dos estrategias de hibridación, una simple (HS) y una asistida (HA). La primera plantea partir buscando una solución con AG, la cual al no presentar

mejoras durante un cierto número de iteraciones (generaciones en AG), rescata al mejor individuo alcanzado hasta el momento y lo utiliza como entrada a SA, quien finalmente explora mejores soluciones. La segunda estrategia también comienza aplicando AGs y, al igual que la estrategia anterior, pasado un número de generaciones sin mejora, activa el componente de SA. SA recibe como entrada un individuo y le retorna al AG una cantidad de soluciones (no solamente la mejor solución, sino que además una parte de todas aquellas soluciones que aceptó en su ejecución) que pasarán a reemplazar parte de la población y provocarán la variabilidad de la misma. Este ciclo se repite hasta que AG completa una cantidad límite de generaciones.

Los parámetros utilizados para el algoritmo genético se muestran en la Tabla 1.

Tabla 1. Parámetros usados en el algoritmo genético.

Parámetros fijos	
Número generaciones	1000
Elitismo	10 %
Cruzamiento	70 %
Parámetros variables	
Mutación	2 %, 4.5 %, 6 %
Tamaño población	20, 50, 100 individuos

Y los parámetros utilizados para simulated annealing se muestran en la Tabla 2. En esta tabla, GSM (Generaciones sin mejora) corresponde a la condición de activación del proceso de simulated annealing, a partir del número de generaciones del algoritmos genético en que no se evidencian mejoras.

Tabla 2. Parámetros usados para simulated annealing.

Parámetros fijos	
α (factor de enfriamiento)	0.95
η (factor de reannealing)	1.5
Número de iteraciones	40
Parámetros variables	
Temperatura inicial	50, 100, 150, 200
GSM	100, 500

La siguiente secuencia de imágenes permite mostrar cómo evoluciona el proceso de corrección, a partir de la imagen a la que se le han superpuesto las líneas correspondientes al *template* creado usando AGs.

Dados los resultados de HA y HS, se tiene que ambos enfoques son capaces de provocar mejoras tanto en fitness como en tiempo por sobre los resultados

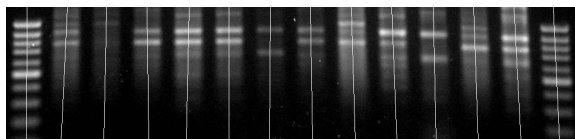


Figura 4. Template producido por la acción del algoritmo genético.

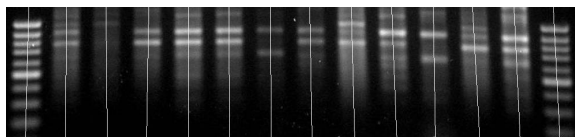


Figura 5. Template modificado por la acción de la hibridación asistida.

obtenidos con AG, y la magnitud de esta mejora varía dependiendo de la combinación de valores que tomen los parámetros en cuestión, razón por la cual a veces los mejores resultados se asocian a HA y otra veces a HS.

El conjunto de pruebas consideró una familia de 13 imágenes, seleccionadas como elementos representativos de una muestra disponible de 150 imágenes. Esto quiere decir que, a juicio de un experto, las imágenes seleccionadas contienen todos los atributos que resultan de interés en todas las imágenes de la muestra.

Aplicando HA, las mejoras se obtuvieron para un 85% de la muestra de imágenes, mientras que con HS se obtuvieron mejoras para un 62%. Un aspecto importante a señalar tiene que ver con la forma en que se fue alcanzando la mejora final de la función objetivo en cuestión: Con HA, la totalidad del valor de la función objetivo fue creciendo a lo largo de casi la totalidad de sus iteraciones. No así con HS, por el cual la totalidad de la función objetivo se enfoca en la partida, y en iteraciones posteriores crece pero más lentamente.

5. Conclusiones

El tratamiento de corrección de imágenes RAPD a través del uso de algoritmos evolutivos, ya se había enfrentado anteriormente con una hibridación entre

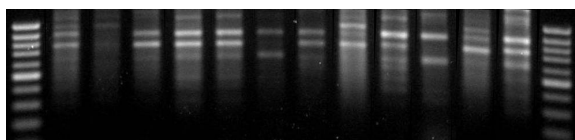


Figura 6. Imagen corregida en base al template generado.

algoritmos genéticos y la técnica de tabu search. El trabajo que aquí se ha desarrollado pretende ampliar el análisis en el uso de mecanismos evolutivos para resolver este problema. Tanto el tratamiento con algoritmos genéticos operando aisladamente, como la hibridación de algoritmos genéticos, ya sea con simulated annealing o con tabu search, se encuentran todavía en fase experimental.

La propuesta aquí explicada permite automatizar el proceso de corrección, con una disminución en tiempo que va desde 8 horas a los 25 minutos (tiempos aproximados), para el tratamiento de una sola imagen. Más allá del tiempo, la calidad de los resultados es el aspecto relevante a considerar, y lo que justifica el esfuerzo en la búsqueda de alternativas efectivas y replicables.

Simulated annealing es capaz de producir mejoras sobre los resultados obtenidos con AG, particularmente con el caso de HA, que cumple con la tarea de sacar constantemente a AG de eventuales óptimos locales. Para la corrección de imágenes RAPD se utilizan las configuraciones asociadas a HA puesto que este presentó mejoras para una mayor cantidad de imágenes que con HA.

Cabe señalar que uno de los factores más complejos de definir fue la temperatura, puesto que fue necesario construir un mecanismo para calcular la temperatura asociada a cada imagen según sus características. Esto último destaca la importancia que posee lograr definir aspectos que caractericen a una imagen, pues en base a eso se pueden generalizar formas de buscar mejoras para distintos casos que se presenten.

Referencias

1. Cao, W., Scoles, G., Hucl, P., Chibbar, R.: Phylogenetic Relationships of Five Morphological Group of Hexaploid wheat Based on RAPD Analysis. *Genome* 43, 724-727 (2000)
2. Casasoli, M., Mattioni, C., Cherubini, M., Villani, F.: A Genetic Linkage Map of European Chestnut (*Castanea Sativa* Mill.) Based on RAPD, ISSR and Isozyme Markers. *Theoretical Applied Genetics* 102, 1190-1199 (2001)
3. Floreano, D., Mattiussi, C.: *Bio-Inspired Artificial Intelligence. Theories, Methods, and Technologies*. The MIT Press (2008)
4. Groos, C., Gay, G., Perrenant, M., Gervais, L., Bernard, M., Dedryver, F., Charmet, G.: Study of the Relationships Between Pre-harvest Sprouting and Grain Color by Quantitative Trait Loci Analysis in the White X Red Grain Bread-wheat Cross. *Theoretical Applied Genetics* 104, 39-47 (2002)
5. Herrera, R., Cares, V., Wilkinson, M., Caligaris, D.: Characterization of Genetic Variations Between *Vitis vinifera* Cultivars from Central Chile Using RAPD and Inter Simple Sequence Repeat Markers. *Euphytica* 124, 139-145 (2002)
6. Muñoz, A.: *Algoritmos Genéticos en la Corrección de Imágenes RAPD*. Informe Técnico, Departamento de Informática, Facultad de Ingeniería, Universidad de Concepción, Chile (2007)
7. Pinninghoff, M. A., Contreras, R., Rueda, L.: An Evolutionary Approach for Correcting Random Amplified Polymorphism DNA Images. *Bioinspired Applications in Artificial and Natural Computation. Lecture Notes in Computer Science* 5602, pp. 467-477, Springer (2009)

8. Rueda, L., Uyarte, O., Valenzuela, S. and Rodriguez, J.: Processing Random Amplified Polymorphism DNA Images Using the Radon Transform and Mathematical Morphology. M. Kamel and A. Campilho (Eds.): ICIAR 2007, LNCS 4633, pp. 1071-1081 (2007)
9. Saal, B., Struss, D.: RGA-and RAPD-derived SCAR Markers for a Brassica B-Genome Introgression Conferring Resistance to Blackleg Oil Seed in Oil Seed Rape. *Theoretical Applied Genetics* 111, 281-290 (2005)
10. Sudapak, M., Akkaya, M., Kence, A.: Analysis of Genetic Relationships Among Perennial and Annual Cicer Species Growing in Turkey Using RAPD Markers. *Theoretical Applied Genetics* 105, 1220-1228 (2002)
11. Tripathi, S., Mathish, N., Gurumurthi, K.: Use of Genetic Markers in the Management of Micropropagated Eucalyptus Germplasm. *New Forests* 31, 361-372 (2006)

Descripción del borde utilizando medidas de compacidad aplicado al diagnóstico automático de cáncer de piel

Raul Santiago-Montero, Noemí Barajas and J. L. Fernando Suárez Y Gómez

Instituto Tecnológico de León, León, Gto,
México
rsantiago66@gmail.com

Resumen Medir la irregularidad del borde en regiones digitales ha despertado gran interés en la comunidad científica. Aplicaciones en el área médica, visión por computador, geometría digital colocan como un rasgo importante la descripción cuantitativa del borde de las regiones. Una aplicación donde se vuelve indispensable la cuantificación de esta propiedad en la detección temprana del melanoma, que es el tipo de cáncer de piel más letal; tres de cada 4 muertes por cáncer de piel se le atribuyen al melanoma. En el presente trabajo se presenta una propuesta para medir la irregularidad del borde de regiones digitales utilizando el concepto de compacidad. Muchas medidas de compacidad se han planteado anteriormente, pero solo algunas han probado una eficaz relación entre la irregularidad del borde y su valor cuantitativo. En el presente trabajo se muestran un conjunto de medidas de compacidad aplicadas a medir la irregularidad del borde de regiones digitales de lesiones de piel. Las descripciones obtenidas general un vector n-dimensional que es utilizado sobre un modelo de inferencia automática que genera un diagnóstico de la lesión; mostrando la eficacia de la propuesta.

Palabras clave: Compacidad, análisis de forma, geometría digital, reconocimiento de patrones.

1. Introducción

El cáncer de piel tiene tasas de crecimiento alarmantes por encima 400% según el reporte anual sobre cáncer [1]. En México, el Instituto Nacional de Cancerología estima el crecimiento en 500 % [1,2].y ocupa el tercer lugar entre los cánceres de la piel a nivel mundial [2]. Dentro de los cánceres de piel, el melanoma es el cáncer más mortal con el 80 % de las muertes y solo el 14 % de los pacientes con melanoma tiene un pronóstico de vida de más de 5 años. [3].

El único tratamiento para el melanoma es la extirpación de la lesión en etapas tempranas. Una vez que el melanoma se desarrolla es altamente resistente a tratamientos, tanto de quimio como radio-terapia y las expectativas de vida para el paciente son escasas. Es por ello la alta prioridad en la detección de este tipo de lesiones en piel apenas su aparición [1-3].

El diagnóstico de melanoma se sospecha por clínica, es decir, si la lesión cumple con ciertas características clínicas que la hagan sospechosa de ser maligna mediante la

dermatoscopia u otros métodos no invasivos y se confirma el diagnóstico con el estudio de las células del tejido enfermo usando procesos de histopatología [3].

La metodología no invasiva más usada para el diagnóstico de melanoma es la regla clínica ABCDE (A= Asimetría, B= Bordes irregulares, C= Color irregular, D= Diámetro, E= Evolución; este último fue introducido para describir los cambios en la forma de las lesiones a través del tiempo) [4-6]. El método se basa en evaluar un conjunto de rasgos o características geométricas y de color o textura de la lesión.

Dado el alto crecimiento en la incidencia del melanoma, el enorme número de regiones susceptibles a ser evaluadas que presenta el cuerpo humano (se estima en al menos 15) y la factible de cuantificar las características; los sistemas de diagnóstico asistido por computadora han despertado un enorme interés [5]. Se ha mostrado que estos sistemas tienen el mismo porcentaje de efectividad que el diagnóstico por clínica por dermatólogos con 2 o más años de experiencia [7]. Existe una cantidad significativa de propuestas que pueden ser divididas en tres segmentos: procesos automáticos de adquisición de imágenes de lesiones, procesamiento de las imágenes y análisis e inferencia [5].

La eficacia en la inferencia del sistema está determinada por la naturaleza de la descripción de la característica que se pretende medir. El presente trabajo se enfoca en la descripción cuantitativa de la característica B de la lesión. Esta característica intenta describir el borde mediante alguna relación o expresión que correlacione la irregularidad del borde con un valor específico.

Existen una cantidad de artículos que detallan el proceso de encontrar una medida para la irregularidad del borde de una lesión en piel. Dos caminos para llegar a una medida de irregularidad han sido el índice de compacidad y la dimensión fractal. El índice de compacidad es el más utilizado por su facilidad de calcular [5, 8, 9].

Sin embargo, se argumenta que es inapropiado para describir el borde de la lesión. En general, el argumento se sustenta en dos hechos: la poca sensibilidad a evaluar la presencia de picos y concavidades, y su fácil alteración a la presencia de ruido [8, 9].

Otra propuesta para medir la irregularidad del borde consiste en dividir de la región de la lesión utilizando dos pares de ejes ortogonales que pasen por el centro de masa de ésta. Una variante de este proceso se describe en [6, 9].

En el 2003, Lee propuso los índices de irregularidad. Estos índices son obtenidos a partir de una transformación de la región bidimensional y obtener una función continua. La transformación utiliza de los puntos en el contorno y mide las diferencias que existen frente a una curva patrón [8].

El presente trabajo se enfoca en obtener una descripción cuantitativa del borde de una lesión en piel mediante medidas basadas en el concepto de compacidad de las formas y diseñadas para el espacio discreto. Este conjunto de medidas permiten generar un vector de características 4-dimensional que es sometido a proceso de clasificación y se obtengan diagnósticos sobre la presencia de un melanoma utilizando solo la característica B de la regla ABCD.

El resto del trabajo está dividido por la descripción de las herramientas utilizadas, el modelo propuesto, los resultados obtenidos y finaliza con una serie de conclusiones.

2. Compacidad y descripción del borde de una lesión

En esta sección se describen las propuestas utilizadas para cuantificar el concepto de compacidad. Además se presenta el modelo de descripción a través de un vector 3-dimensional que será la materia básica en el proceso de automatización del diagnóstico.

2.1. Medidas de compacidad

El concepto de compacidad hace referencia al grado de densidad-porosidad de un objeto o al grado disperso-compacto de la forma de este objeto [10]. En el caso de la forma, el concepto se aplica tanto al plano como al espacio. Cuantificar el concepto ha recaído sobre la teoría isoperimétrica desde hace varias décadas [11] y su relación base se ha aplicado como medida de compacidad de las formas [11-15], que se determina como:

$$C = \frac{P^2}{A} \quad (1)$$

No obstante su amplio uso, se detectaron efectos no deseables cuando la medida es aplicada sobre formas contenidas en imágenes digitales [8,16, 15]. La medida, en ámbito de las imágenes digitales, es afectada de manera importante por la resolución y la rotación de la forma bajo análisis. Esto es, la forma de un objeto contenida en una imagen digital a diferente resolución produce diferentes valores de compacidad, bajo la medida isoperimétrica. Esta inconsistencia se incrementa cuando se agrega una rotación a la forma.

Con el incremento y uso de imágenes digitales en diversas aplicaciones, el generar una medida adecuada para la compacidad se ha convertido, en si mismo, en un problema de investigación. Hasta ahora se encuentran reportadas 12 propuestas de medir el concepto de compacidad sobre formas en imágenes digitales [17,18]. De todas ellas, 4 propuestas se han tomado en función de su sencillez de implementación y que realizan la evaluación de la compacidad aplicando el enfoque de similitud entre la forma bajo estudio y una forma referencia; mismo enfoque que presenta la medida clásica de compacidad.

La primera propuesta que incluimos en este trabajo es la medida de circularidad de Haralick [19]. Esta medida determina la compacidad de una región utilizando el conjunto de distancias entre el centroide de la región y las celdas en su borde. Una vez que es determinado el conjunto de distancias, se obtiene un valor de compacidad mediante la media de las distancias entre su desviación estándar. No obstante el mayor proceso de cómputo de la medida, la medida de circularidad de Haralick se muestra sensible a los cambios en la morfología de región. El primer paso para determinar la medida es calcular el centroide de la región digital. Shapiro define el centroide como la localización promedio de los elementos de una región digital [20], denotado como (\bar{r}, \bar{c}) donde:

$$\bar{r} = \sum_{(r,c) \in R} r \quad (2)$$

$$\bar{c} = \sum_{(r,c) \in R} c \quad (3)$$

Donde r y c son las posiciones en renglón y columna de las celdas de la región digital. A pesar de que los renglones y columnas son números enteros, el centroíde frecuentemente resulta en una posición en el espacio continuo, por lo tanto, se ubicará en la celda más próxima a su valor de renglón y de columna.

A partir de este punto de la región digital se calculan distancias euclidianas hacia cada celda en el borde de la región. La medida de circularidad de Haralick queda entonces expresada como:

$$C = \frac{\mu_R}{\sigma_R} \quad (4)$$

donde:

$$\mu_R = \frac{1}{K} \sum_{k=0}^{K-1} \|(r_k, c_k) - (\bar{r}, \bar{c})\| \quad (5)$$

y

$$\sigma_R = \left(\frac{1}{K} \sum_{k=0}^{K-1} (\|(r_k, c_k) - (\bar{r}, \bar{c})\|)^2 \right)^{1/2} \quad (6)$$

Por otra parte, Danielsson utiliza un enfoque diferente en el diseño de una medida que evalué la compacidad de las regiones digitales. El propone usar la información que proporciona cada celda a través de la transformada de distancia [21]. Al aplicar la transformada de distancia, Danielsson utiliza el principio del momento de inercia y lo normaliza para que el círculo produzca el valor mínimo. La propuesta es nombrada factor G y se expresa, para el espacio discreto, como:

$$G = \frac{N^3}{9\pi(\sum_1^N x_i)^2} \quad (7)$$

Donde N es el número de celdas en la región digital y x_i es el valor de la transformada de distancias en cada celda de la región.

La última medida de compacidad que se emplea en este trabajo es la propuesta por Peura [22]. Esta medida establece una relación entre dos regiones. La relación compara la región digital bajo estudio contra un círculo en el espacio continuo. La relación de circularidad de Iivarién y Peura se expresa como una relación entre el perímetro de la región digital y un círculo con igual área. La medida de circularidad de Peura está dada por:

$$C = \frac{2\sqrt{A\pi}}{P_D} \quad (8)$$

Donde A es el número de celdas de la región digital y P_D es el perímetro de la región digital.

2.2. Knn

El K-vecino más cercano (K-nearest neighbor) es un método de estimación no paramétrica muy bien conocido en el área de reconocimiento de patrones bajo representaciones vectoriales n-dimensionales de objetos a clasificar. Este método es particularmente efectivo cuando la distribución de probabilidad del conjunto de patrones es desconocida [23].

Entonces, sea L el número de clases, N el número total de patrones, N_i el número de patrones pertenecientes a cada clase, conocidos como prototipos. Sea Φ el conjunto de los prototipos y donde $y \in \Phi$ es un vector de dimensión n .


La inferencia o clasificación del KNN se determina por una función $f(x, y)$ y es una función no negativa y simétrica; donde el valor más alto denota la mayor similitud entre un el vector y y el vector x . Es común utilizar la función de distancia euclidiana como función de similitud.

Este método se emplea en el presente trabajo para generar un diagnóstico, sobre la lesión presente en una imagen digital binaria, entre melanoma (maligna) y una lesión benigna.

2.3. Descripción del borde de la lesión por compacidad


Las lesiones que se emplean son: imágenes segmentadas, binarizadas y diagnosticadas previamente. La descripción se realiza median un vector 3-dimensional de compacidad.

Tabla 1. Valores obtenidos para una lesión diagnosticada por dermatólogos como benigna.

Region benigna	Descripción
	0.731971643857869
	19.08214965799585
	1.687351050412136

En la Tabla 1 se obtiene la descripción de una lesión clasificada como benigna a través de un vector 3-dimensional de compacidad utilizando las medidas descritas y empleadas en este estudio; donde el primer valor es el obtenido por (8), la segunda por (4) y finalmente, se muestra el valor obtenido por (7). Por otra parte, la Tabla 2 muestra la descripción para una lesión diagnosticada como maligna o melanoma.

Tabla 2. Descripción obtenida por las medidas incluidas en el estudio para una lesión maligna

Región maligna (melanoma)	Descripción
	0.5319088810047378
	5.7119639900771855
	2.708315581066739

3. Resultados

En este estudio se utilizaron un conjunto de 20 imágenes binarias de las regiones digitales que contienen una lesión en piel humana. Las imágenes están clasificadas previamente y divididas en dos categorías, diez son benignas y 10 malignas. Del conjunto de imágenes se seleccionaron 5 con regiones clasificadas como benignas y 5 como malignas como conjunto de entrenamiento del modelo KNN.

Tabla 3. Regiones usadas como conjunto de entrenamiento del clasificador KNN en el diagnóstico de melanoma.

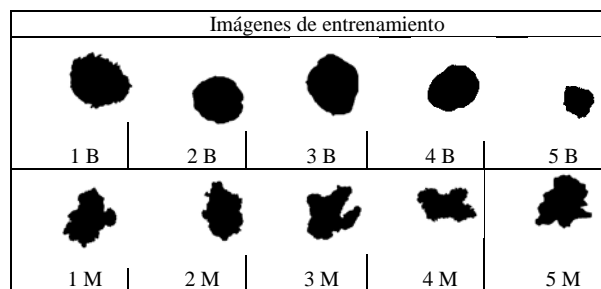


Tabla 4. Descripción de las regiones benignas mostradas en el primer renglón de la Tabla 3 mediante sus valores de compacidad.

Datos de entrenamiento: benignos			
Imagen	Peura	Haralick C	Factor G-
1	0.601	10.4558	1.894
2	0.731	19.0821	1.687
3	0.743	14.51696	1.769
4	0.752	10.5891	1.681
5	0.696	10.15409	1.527

En la Tabla 3 se muestran las regiones digitales binarizadas las lesiones de este conjunto y en la Tabla 4 y 5, sus vectores de características en función de las medidas de compacidad.

Las imágenes restantes se propusieron como imágenes de prueba, las cuales se presentan en la Tabla 6 y 7 con sus diagnósticos previos y el resultado arrojado por el modelo de clasificación en el último renglón. La eficacia de la representación propuesta de compacidad se puede observar en la Tabla 8 que muestra la matriz de confusión del experimento realizado.

4. Conclusiones

No obstante lo limitado de las muestras de los conjuntos de entrenamiento y prueba, los resultados obtenidos por los experimentos realizados muestran que la propues-

ta puede realizar una efectiva clasificación de lesiones de piel humana y de generar un diagnóstico del cáncer de piel, conocido como melanoma. Hay que destacar que la propuesta solo emplea una característica de la regla ABCD para el diagnóstico de melanoma empleada por expertos en cáncer de piel.

Tabla 5. Descripción de las regiones malignas ilustradas en el segundo renglón de la Tabla 3.

Datos de entrenamiento: malignos			
Imagen	Peura	Haralick C	Factor G
1	0.53190	5.711963	2.708319
2	0.58264	5.97185	2.09514
3	0.48831	4.61046	3.314071
4	0.4785	4.00625	3.07477
5	0.579341	6.30134	2.54245

Tabla 6. Conjunto de imágenes de melanoma, su descripción mediante compacidad y el resultado obtenido por el clasificador junto a un diagnóstico previo.






Imagen					
Peura	0.4601	0.5325	0.5195	0.4365	0.5663
Haralick	5.8355	3.3935	4.5333	5.6488	6.2934
Factor G	2.7250	2.4890	2.9476	2.3757	2.4715
Resul	M-M	M-M	M-M	M-M	M-M

Tabla 7. Conjunto de imágenes de lesiones de piel benignas, su descripción mediante compacidad y el resultado obtenido por el clasificador junto a un diagnóstico previo.






Imagen					
Peura	0.60443	0.60713	0.72447	0.71845	0.76838
Haralick	16.48476	12.69999	11.17549	9.19046	14.66675
Factor G	1.79331	1.74325	1.72806	1.88039	1.71360
Resul	B-B	B-B	B-B	B-B	B-B

Table 8. Matriz de confusión resultante del proceso de clasificación.

Lesión	Benigna	Maligna
Benigna	5	0
Maligna	0	5

Además se muestra que el concepto de compacidad es útil en la generación de una descripción del borde de una región digital. Las medidas aquí empleadas evitan los efectos nocivos que investigadores detecta en la medida clásica de compacidad.

El presente trabajo abre un campo para probar la sencillez, eficacia y lo robusto del concepto de compacidad en diversas aplicaciones donde el borde de las regiones sea un factor necesario para procesos de evaluación y clasificación.

Referencias

1. R.M. Gutierrez Vidrio y N. Cortés Lozano: Confrontando al melanoma en el siglo XXI. *Med Cutan Iber Lat Am*, 35(1), pp. 3-13 (2007)
2. A. De la Fuente-García y J. Ocampo-Candiani: Melanoma cutáneo. *Gac. méd. Méx.*, 146(2), pp. 126-135 (2010)
3. A. Alfaro, L. Castrejón y M. Rodríguez Ortíz: Cáncer de piel. Estudio epidemiológico a 10 años de derechohabientes del ISSSTE en Nuevo León. *Dermatología Rev Mex*, 54(6), pp. 321-325 (2010)
4. M. d'Amico, M. Ferri, and I. Stanganelli, I.: Qualitative asymmetry measure for melanoma detection. In: *ISBI2004: Proceedings of the IEEE International Symposium on Biomedical Images* (Arlington, VA) (2004)
5. I. Maglogiannis, C. N. Doukas: Overview of Advanced Computer Vision Systems for Skin Lesions Characterization. *Information Technology in Biomedicine, Trans on.*, 13(5), pp. 721-733 (2009)
6. A. Bono et al.: The ABCD System of Melanoma Detection, A spectrophotometric Analysis of the Asymmetry, Border, Color, and Dimension. *Cancer*, 85(1), pp. 72-77 (1998)
7. S.M. Rajpara, A. P. Botello, J. Townend y A. D. Ormerod: Systematic review of dermoscopy and digital dermoscopy/ artificial intelligence for the diagnosis of melanoma. *British Journal of Dermatology*, 161, pp. 591-604 (2009)
8. T.K. Lee and Ela Claridge: Predictive power of irregular border shapes for malignant melanomas. *Skin Research and Technology*, 11, pp. 1-8 (2005)
9. Tor-Björn Holmström: A Survey and Evaluation of Features for the Diagnosis of Malignant Melanoma. *Mater Thesis* (2005)
10. Real Academia Española: *Diccionario de la lengua española*. Ed. 23. [11] R. O.
11. K.R. Castleman: *Digital Image Processing*. Prentice Hall (2010)
12. L.F. Costa and R. M. Cesar: *Shape Analysis and Classification: Theory and Practice*. CRC Press, Boca Raton, Florida, USA, (2009)
13. R.C. Gonzalez and R. E. Woods: *Digital Image Processing*. Prentice Hall, Upper Saddle, New Jersey, USA (2007)
14. A.K. Jain: *Fundamentals of Digital Image Processing*. Prentice Hall, USA (1989)
15. A. Rosenfeld: Compact figures in digital pictures. *Systems, Man and Cybernetics, Trans on.*, 4, pp. 221-223 (1974)
16. M.J. Bottema: Circularity of objects in images. In: *International Conference on Acoustic, Speech and Signal Processing, ICASSP 2000, Istanbul*, pp 2247-2250, (2000).
17. N. Ritter and James Cooper: New Resolution Independent Measures of Circularity. *J Math Imaging Vis*, 35, pp. 117-127 (2009)
18. Santiago Montero, R. and Ernesto Bribiesca: State of the art of compactness and circularity measures. *International of Mathematical Forum*, 4(27), pp. 1305-1335 (2008)
19. R.M. Haralick: A measure for circularity of digital figures, *Systems. Man and Cybernetics, Trans on.*, SMC- 4(4), pp. 334-336 (1974)
20. L.G. Shapiro and G. C. Stockman: *Computer Visión*. Prentice Hall, Upper Saddle River, New Jersey, USA (2001)
21. P.E. Danielsson: A new shape factor. *Computer Graphics and Image Processing*, 7(2):292-299 (1978)

22. M. Peura and J. Iivarinen: Efficiency of simple shape descriptors. In: C. Arcelli, P. Cordella and G Sanniti de Baja, editors, *Advances in Visual Form Analysis*, pp. 443-451, World Scientific, Singapore (1997)
23. E. Fix and J. L. Hodges Jr.: Discriminatory Analysis: Nonparametric Discrimination Consistency Properties. Project 21-49-004, Report Number 4, USAF School of Aviation Medicine, pp 261-279 (1951)

Asimetría de regiones digitales bajo el concepto de compacidad en el diagnóstico de cáncer de piel

Raul Santiago, Antonio Zamarrón y Angélica Ortiz-Gaucin

Instituto Tecnológico de León, León, Gto,
México
rsantiago66@gmail.com

Resumen El cáncer de piel tiene tasas de crecimiento del 400% a nivel mundial y en México este porcentaje es más alto. Entre todos los cánceres detectados en el mundo, uno de cada tres corresponde a la categoría de cáncer cutáneo. Entre todos los cánceres de piel, el más letal es el melanoma. Detectar en etapas tempranas y de forma masiva se ha convertido en un tema de interés mundial. La regla ABCD es el principal instrumento de que se sirven los especialistas, de manera cualitativa, en el diagnóstico temprano de este padecimiento. La A se refiere al grado de asimetría que presenta la lesión, B hace alusión al borde, C a las tonalidades presentes y D al tamaño. Los sistemas de diagnóstico asistido por computadora se han convertido en una opción para ayudar al diagnóstico masivo y en donde el principal componente es la descripción de estas propiedades de las lesiones. En el presente trabajo se hace uso de la geometría digital para generar una descripción de la primera característica, la asimetría, utilizando el concepto de compacidad para generar un vector de características cuatro-dimensional. Esta propuesta se prueba con varias medidas que ilustran la utilidad del concepto en procesos de diagnóstico asistido mediante procesos de clasificación automática.

Palabras clave: Geometría digital, reconocimiento de patrones, análisis de forma, compacidad discreta.

1. Introducción

Uno de los cánceres más agresivos es el melanoma. El melanoma es una clase de cáncer de piel generado a partir de melanocitos atípico. Este tipo de cáncer se caracteriza por su gran capacidad de producir metástasis. La incidencia de este tipo de cáncer se está convirtiendo en un problema de alcances mundiales. La Organización Mundial de la Salud reporta que uno de cada tres pacientes diagnosticado con cáncer tiene algún tipo de cáncer de piel [1]. Del total de casos de cáncer de piel, el cuatro por ciento es melanoma. Acorde a las estadísticas del Instituto Mexicano de Cancerología, en México la incidencia ha aumentado en 500% [2].

El enorme crecimiento en la incidencia de cáncer de piel y la alta agresividad del melanoma ha despertado el interés de la comunidad científica por el desarrollo de sistemas automáticos de detección temprana de melanoma [3, 4]. Hasta el 2009 se habían escrito más de 400 estudios sobre descripción para el diagnóstico a través de

sistemas asistido por computadora [4]. La caracterización de una lesión sospechosa de ser cáncer de piel se realiza principalmente por la regla ABCD, donde A se refiere a la Asimetría de la lesión, B a la irregularidad del Borde, C a la diferencia de tonalidad en la lesión y D es la dimensión de la lesión.

Para el caso de la asimetría, ésta se evalúa a partir de un punto dentro de la región. Una vez localizado en punto se divide la región de la lesión por uno o más ejes. Las subregiones obtenidas a partir de los ejes generados se someten a diversos métodos de evaluación. Entre los métodos más generalizados esta la correlación de puntos entre las subregiones obtenidas por un traslape de estas subregiones [3, 5, 6].

El presente trabajo se propone un proceso para evaluar la asimetría de las lesiones de piel utilizando el concepto de compacidad. La compacidad hace alusión al grado de dispersión que un objeto o figura plana muestra. Existen más de 20 propuestas para medir este concepto en el espacio continuo. No obstante, estas propuestas presentan inconvenientes cuando son aplicadas sobre imágenes digitales, que hoy en día son la naturaleza donde se capturan las imágenes de las lesiones en piel. Es por ello que se aplican medidas de compacidad diseñadas específicamente para el espacio digital.

Con las medidas de compacidad en el espacio digital, se propone un vector de características 4 dimensional que es presentado a un proceso de inferencia automático; en nuestro caso hemos elegido el algoritmo KNN, por su sencillez y alto grado de eficacia. Los vectores de características de cada medida son comparados y es medida la eficacia de cada descripción, así como de la propuesta.

EL resto del documento se organiza como sigue: en la sección dos se describe el concepto de compacidad y las medidas de compacidad para el espacio digital utilizadas en el proceso comparativo. En la sección tres se muestra la propuesta de descripción. En la sección cuatro se presentan los resultados obtenidos y finalmente se presentan algunas conclusiones.

2. Compacidad

El concepto de compacidad hace referencia al grado de densidad-porosidad de un objeto o al grado disperso-compacto de la forma de este objeto [7]. En el caso de la forma, el concepto se aplica tanto en el plano como en el espacio. Cuantificar el concepto ha recaído sobre la teoría isoperimétrica desde hace varias décadas [8] y su relación base se ha aplicado como medida de compacidad de las formas [8-12], que está dada por:

$$C = \frac{P^2}{A} \quad (1)$$

No obstante su amplio uso, se detectaron efectos no deseables cuando la medida es aplicada sobre formas contenidas en imágenes digitales [13-15, 14]. La medida, en ámbito de las imágenes digitales, es afectada de manera importante por la resolución y la rotación de la forma bajo análisis. Esto es, la forma de un objeto contenida en una imagen digital a diferente resolución produce diferentes valores de compacidad, bajo la medida isoperimétrica. Esta inconsistencia se incrementa cuando se agrega una rotación a la forma.

Con el incremento y uso de imágenes digitales en diversas aplicaciones, el generar una medida adecuada para la compacidad se ha convertido, en si mismo, en un problema de investigación. Hasta ahora se encuentran reportadas 12 propuestas de medir el concepto de compacidad sobre formas en imágenes digitales [16, 17]. De todas ellas, 4 propuestas se han tomado en función de su sencillez de implementación y que realizan la evaluación de la compacidad aplicando el enfoque de similitud entre la forma bajo estudio y una forma referencia; mismo enfoque que presenta la medida clásica de compacidad.

La primera propuesta que describimos fue diseñada por Bribiesca y se apega al comportamiento que tiene la medida isoperimétrica en espacio continuo. La medida recibió el nombre de Compacidad Discreta Normalizada [18] y ha mostrado su utilidad para evaluar la evolución de tumores cancerígenos; Braumman et al. utilizan como único parámetro los valores de la C_{DN} en tumores cervico-uterinos [19]. La C_{DN} emplea una relación entre representaciones lineales de áreas. Esta relación compara el área de la región digital de la forma bajo estudio contra una región digital cuadrada con igual área a través de su compacidad discreta C_D . Los valores de la C_{DN} están contenidos en el rango 0-1; mientras que las formas compactas producen valores cercanos a 1 y las más dispersas tenderán al valor de 0. Esta relación se expresa como:

$$C_{DN} = \frac{C_D - C_{Dmin}}{C_{Dmax} - C_{Dmin}}. \quad (2)$$

donde

$$C_{D=\frac{4n-P}{2}} \quad (3)$$

donde n es el número de pixel que contiene la región digital y P es el perímetro de la región. C_{Dmin} es la compacidad discreta mínima de una región digital con n pixel y está dada por:

$$C_{Dmin} = n - 1. \quad (4)$$

y C_{Dmax} es la compacidad discreta máxima para una región compuesta por n pixels y se expresa como

$$C_{Dmax} = \frac{4n - 4\sqrt{n}}{2}. \quad (5)$$

Otra propuesta que incluimos en este trabajo es la medida de circularidad de Haralick [20]. Esta medida determina la compacidad de una región utilizando el conjunto de distancias entre en centroide de la región y las celda en su borde. Una vez que es determinado el conjunto de distancias, se obtiene una valor de compacidad mediante la media de las distancias entre su desviación estándar. No obstante el mayor proceso de cómputo de la medida, la medida de circularidad de Haralick se muestra sensible a los cambios en la morfología de región. El primer paso para determinar la medida es calcular el centroide de la región digital. Shapiro define el centroide como la localización promedio de los elementos de una región digital [21], denotado como (\bar{r}, \bar{c}) , donde:

$$\bar{r} = \sum_{(r,c) \in R} r, \quad (6)$$

$$\bar{c} = \sum_{(r,c) \in R} c. \quad (7)$$

Donde r y c son las posiciones en renglón y columna de las celdas de la región digital. A pesar de que los renglones y columnas son números enteros, el centroíde frecuentemente resulta en una posición en el espacio continuo, por lo tanto, se ubicará en la celda más próxima a su valor de renglón y de columna.

A partir de este punto de la región digital se calculan distancias euclidianas hacia cada celda en el borde de la región. La medida de circularidad de Haralick queda entonces expresada como:

$$C = \frac{\mu_R}{\sigma_R}. \quad (8)$$

donde

$$\mu_R = \frac{1}{K} \sum_{k=0}^{K-1} \|(r_k, c_k) - (\bar{r}, \bar{c})\|, \quad (9)$$

y

$$\sigma_R = \left(\frac{1}{K} \sum_{k=0}^{K-1} (\|(r_k, c_k) - (\bar{r}, \bar{c})\|)^2 \right)^{1/2}. \quad (10)$$

La última medida de compacidad que se emplea en este trabajo es la propuesta por Peura [25]. Esta medida, al igual que la CDN, establece una relación entre dos regiones. No obstante, la relación compara la región digital bajo estudio contra un círculo en el espacio continuo, en vez de una región digital cuadrada como en el caso de la CDN. La relación de circularidad de Iivarién y Peura se expresa como una relación entre el perímetro de la región digital y un círculo con igual área. La medida de circularidad de Peura está dada por:

$$C = \frac{2\sqrt{A\pi}}{P_D} \quad (11)$$

Donde A es el número de celdas de la región digital y P_D es el perímetro de la región digital.

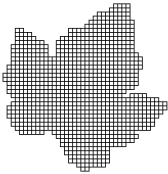
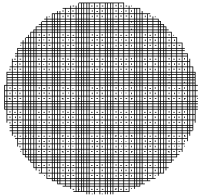
En la Tabla 1 se muestran dos regiones digitales compuestas por celdas cuadradas y los valores obtenidos por cada una de las medidas de compacidad probadas.

3. Descripción de la asimetría por compacidad

La asimetría de las neoplasias en piel es uno de los parámetros que contribuye de manera importante, bajo la regla ABCD, para el diagnóstico de melanoma maligno. El proceso general para evaluar la asimetría de una lesión en piel consiste en: determinar un par de ejes ortogonales que dividan la forma de la lesión en cuatro secciones y que tengan como origen el centro de masa; se evalúan las diferencias entre las regiones contenidas en cada cuadrante y se determina un grado de asimetría [3, 22].

Ng y Cheung utilizan un la distancia de simetría [6, 23] para establecer el grado de asimetría de la lesión. La distancia de simetría se genera a partir de medir la distancia entre un conjunto de pares de puntos entre la región a evaluar y una figura simétrica de referencia. La medida se aproxima a cero si la región guarda una similitud con la

Tabla 1. Valores obtenidos por las diversas propuestas para medir la compacidad de regiones digitales; (a) forma dispersa; (b) forma compacta.

Región digital	Clásica	Discreta Norm	Haralick	Peura
 <p>(a)</p>	40.01	0.9599	5.44	1.784
 <p>(b)</p>	20.49	0.9999	84.35	1.276

forma simétrica de referencia. Conforme se requiere una mejor evaluación de la asimetría de la región es necesario evaluar más puntos. No obstante para un conjunto relativamente pequeño, de alrededor de cientos de puntos, el proceso eleva significativamente el trabajo de cómputo. La solución propuesta por Ng y Cheung es limitar el número de puntos haciendo un muestreo a intervalos angulares regulares, con el fin de reducir el proceso de cálculo.

Por otra parte, Bono et al. [22] describen la asimetría de una lesión como el grado de similitud entre la lesión y el círculo. El grado de similitud se obtiene mediante (1). Propuestas más sofisticadas se han aplicado en la medición de la asimetría. Tal es el caso propuesto en [5] donde emplean un proceso topológico ubicando el centro de masa de la región y un conjunto de distancias aplicadas a la función de tamaño [24]

La propuesta en [5] trata las regiones digitales como regiones continuas e interpola los resultados. Sin embargo, las regiones digitales y los parámetros obtenidos dependen fuertemente de la resolución de la imagen digital. Esto es: a diferente resolución, diferentes valores.

Santiago et al. [16] utilizan el concepto de centroide, propuesto por Shapiro [21], como punto donde los ejes tengan su origen. A partir de las posiciones de renglón y columna se traza el par de ejes que dividen la región en dos pares. Un par es por el eje vertical y otro par por el eje horizontal

Se realiza una medición usando la CDN y se obtiene un vector 4 dimensional que sirve como la firma de asimetría de cada lesión.

El proceso se ilustra con las regiones de la Figura 3, donde son evaluadas mediante el proceso de asimetría propuesto por Santiago et al. Los valores de compacidad entre regiones deberán tener menor discrepancia si son simétricas y lo contrario para regiones asimétricas.











Región	(a)	(b)	(c)	(d)
				
1	(a)	(b)	(c)	(d)
				
2	(a)	(b)	(c)	(d)

Fig. 1. División de las regiones de la Tabla 1 en cuatro sub-regiones a partir de dos ejes que cruzan por el centro de masa de la región.

En el presente trabajo se realiza un proceso comparativo entre la medida de compacidad discreta normalizada de Bribiesca frente a otras cuatro propuestas de medidas de compacidad [17]. Ahora un vector de características se forma por cada medida de compacidad.

Tabla 2. Descripción de la asimetría de las lesiones en piel representadas en la Figura 3 utilizando diferentes medidas de compacidad. Las celdas resaltadas muestran las diferencias en valores de compacidad más notorias.

<i>Región 1</i>	1(a)	1(b)	1(c)	1(d)
Clásica	36.081818	36.623077	35.135135	32.100209
CDN	0.94725531	0.95079233	0.95175802	0.95821481
C Haralick	4.0552708	2.6715716	3.3698502	3.1490382
C Peura	1.694491	1.7071531	1.672114	1.5982658
<i>Region 2</i>	2(a)	2(b)	2(c)	2(d)
Clásica	22.760536	23.142857	23.29306	23.036649
CDN	0.99118411	0.99055613	0.99034196	0.99088374
C Haralick	4.190369	4.0694204	4.0059821	4.173777
C Peura	1.3458179	1.3570741	1.3614708	1.3539565

La Tabla 2 muestra en cada renglón un vector de características de compacidad para cada región de la Figura 3, obtenidos de cada una de las sub-regiones, ilustradas en la Figura 3. Se espera que una región simétrica obtenga valores entre sus subregiones (a) y (b) o (c) y (d) que produzcan diferencia mínimas entre ella, lo contrario sería para regiones con asimetría. En el ejemplo de la Figura 3 y con los datos de la Tabla 2 se observa que para la región 1 de la Figura 3 siempre existen notorias diferencias entre las subregiones. No obstante que todas las medidas utilizadas en esta propuesta muestran asimetría en la región 1, la medida C de Haralick produce un mejor vector de características para la evaluación de la asimetría; esto al marcar diferencias en los dos pares de regiones divididas por los ejes resultantes.





















<i>Benignas</i>				
 1	 2	 3	 5	 6
<i>B1</i>	<i>B2</i>	<i>B3</i>	<i>B4</i>	<i>B5</i>
 7	 8	 9	 10	 11
<i>B6</i>	<i>B7</i>	<i>B8</i>	<i>B9</i>	<i>B10</i>
<i>Malignas</i>				
 1	 2	 6	 7	 8
<i>M1</i>	<i>M2</i>	<i>M3</i>	<i>M4</i>	<i>M5</i>
 9	 10	 11	 12	 15
<i>M6</i>	<i>M7</i>	<i>M8</i>	<i>M9</i>	<i>M10</i>

Fig. 2. Regiones de entrenamiento (regiones B1-B5 y M1-M5) y de prueba (regiones B6-B10 y M6-M10) para el algoritmo de clasificación KNN.

4. Pruebas experimentales

Para probar la capacidad de evaluar la asimetría bajo el concepto de la compacidad, la descripción de asimetría propuesta se emplea en un proceso de clasificación automática. El proceso de clasificación utiliza el algoritmo estadístico, no-paramétrico y supervisado KNN, con K igual a uno. En nuestro experimento se utilizan las descripciones de las cinco primeras lesiones benignas y las cinco primeras lesiones malignas de la Figura 2. Con este conjunto se configura el conjunto de entrenamiento del clasificador, las cuales ya han sido diagnosticadas previamente. Posteriormente se someten las siguientes las restantes 10 regiones de la Figura 2 para obtener la una matriz de confusión que valora la eficacia de la propuesta.

El experimento se realiza utilizando el algoritmo KNN con una K=1, donde se someten vecotes de características 4-dimensional. Se realiza un proceso de clasificación por cada una de las medidas utilizadas. Las matrices de confusión

mostradas en las Tablas 3, 4, 5 y 6 contienen los resultados de clasificación para cada medida.

Tabla 3. Matriz de confusión para la clasificación de la muestra utilizando la medida clásica de compacidad.

Clásica	Benignas	Malignas	%
Benignas	3	2	60
Malignas	0	5	100

Tabla 4. Resultado de la clasificación por KNN empleando la medida C de circularidad de Peura.

Peura	Benignas	Malignas	%
Benignas	4	1	80
Malignas	2	3	60

Tabla 5. Eficacia de la medida de circularidad de Haralick para el conjunto muestra.

Benignas	2	3	40
Malignas	1	4	80

Tabla 6. Matriz de confusión resultado del proceso de clasificación aplicando la medida de compacidad discreta normalizada.

Discreta Norm.	Benignas	Malignas	%
Benignas	4	1	80
Malignas	1	4	80

5. Conclusiones

La propuesta de evaluación de la asimetría utilizando las medidas de compacidad diseñadas para el espacio digital tienen las características de ser de bajo gasto computacional y simples de implementar. Los resultados muestran que aplicando un proceso de división mediante dos ejes arbitrarios y evaluando las compacidades de cada región puede describirse la asimetría de una región.

El trabajo presenta una aplicación donde la evaluación de esta propiedad resulta ser indispensable, como es el caso de la detección temprana de melanomas. Además, se muestra lo robusto que resulta ser medida de compacidad discreta normalizada.

Aunque los resultados no pueden ser concluyentes para inferir que la descripción de la asimetría propuesta es un indicador para el diagnóstico automático del melanoma humano, si se establece una metodología que puede ser extendida a evaluar las otras propiedades geométricas de las lesiones en piel.

Referencias

1. A. Alfaro, L. Castrejón y M. Rodríguez Ortíz: Cáncer de piel. Estudio epidemiológico a 10 años de derechohabientes del ISSSTE en Nuevo León. *Dermatología Rev Mex*, 54(6), 321-325 (2010)
2. R.M. Gutierrez Vídrio y N. Cortés Lozano: Confrontando al melanoma en el siglo XXI. *Med Cutan Iber Lat Am*, 35(1), pp. 3-13 (2007)
3. I. Maglogiannis, C.N. Doukas: Overview of Advanced Computer Vision Systems for Skin Lesions. Characterization, Information Technology in Biomedicine, *Trans on.*, 13(5), pp. 721-733 (2009)
4. S.M. Rajpara, A.P. Botello, J. Townend y A.D. Ormerod: Systematic review of dermoscopy and digital dermoscopy/ artificial intelligence for the diagnosis of melanoma. *British Journal of Dermatology*, 161, pp. 591-604 (2009)
5. M. d'Amico, M. Ferri, and I. Stanganelli, I.: Qualitative asymmetry measure for melanoma detection. In: *ISBI2004: Proceedings of the IEEE International Symposium on Biomedical Images (Arlington, VA)* (2004)
6. V. Ng and D.W.L. Cheung: Measuring asymmetries of skin lesion. *Computational Cybernetics and Simulation, IEEE International Conference on Systems, Man, and Cybernetics Conference Proceedings, Orlando, Florida, USA, 12-15 October 1997*, v. 5, p. 4211-4216 (1997)
7. Real Academia Española, *Diccionario de la lengua española*, Ed. 23 (2010)
8. R.O. Duda and P.E. Hart: *Pattern Classification and Scene Analysis*. Wiley-Interscience, New York, USA (1973)
9. K.R. Castleman: *Digital Image Processing*. Prentice Hall (1995)
10. L.F. Costa and R.M. Cesar: *Shape Analysis and Classification: Theory and Practice*, CRC Press, Boca Raton, Florida, USA (2009)
11. R.C. Gonzalez and R.E. Woods: *Digital Image Processing*. Prentice Hall, Upper Saddle, New Jersey, USA (2007)
12. A.K. Jain, *Fundamentals of Digital Image Processing*. Prentice Hall, USA (1989)
13. A. Rosenfeld: Compact figures in digital pictures. *Systems, Man and Cybernetics, Trans on.*, 4, pp. 221-223 (1974)
14. T.K. Lee and Ela Claridge: Predictive power of irregular border shapes for malignant melanomas. *Skin Research and Technology*, 11, pp. 1-8 (2005)
15. M.J. Bottema: Circularity of objects in images. In: *International Conference on Acoustic, Speech and Signal Processing, ICASSP 2000, Istanbul*, pp 2247-2250 (2000)
16. Santiago Montero, R. and Ernesto Bribiesca: State of the art of compactness and circularity measures. *International of Mathematical Forum*, 4(27), pp. 1305-1335 (2008)
17. N. Ritter and James Cooper: New Resolution Independent Measures of Circularity. *J Math Imaging Vis*, 35, pp. 117-127 (2009)
18. E. Bribiesca: Measuring 2D Shape Compactness using the Contact Perimeter. *Computer and Mathematics with Applications*, 33(11), pp. 1-9 (1997)
19. U.D. Braumann et al.: Three-dimensional reconstruction and quantification of cervical carcinoma invasión fronts from histological serial section. *Trans on Medical Imaging*, 24(10), pp. 1286-1307 (2005)
20. R.M. Haralick: A measure for circularity of digital figures. *Systems, Man and Cybernetics, Trans on.*, SMC- 4(4), pp. 334-336 (1974)
21. L.G. Shapiro and G.C. Stockman: *Computer Visión*. Prentice Hall, Upper Saddle River, New Jersey, USA (2001)
22. A. Bono et al.: The ABCD System of Melanoma Detection, A spectrophotometric Analysis of the Asymmetry, Border, Color, and Dimension. *Cancer*, 85(1), pp. 72-77 (1998)
23. H. Zabrodsky, Peleg, S. and D. Avnir: Continuous symmetry measure. *J. Am. Chem. Soc.*, 114, pp. 7843-7851 (1992)

24. Frosini P.: Connections between Size Functions and Critical Points. *Math. Meth. Appl. Sci.* 19, pp. 555–569 (1996)
25. M. Peura and J. Iivarinen: Efficiency of simple shape descriptors. In: C. Arcelli, P. Cordella and G Sanniti de Baja, editors, *Advances in Visual Form Analysis*, pp. 443-451, World Scientific, Singapore (1997)

Algoritmos de procesamiento del iris para un sistema de reconocimiento biométrico

Ulises Orozco-Rosas¹, Mireya S. García-Vazquez¹ y Alejandro A. Ramírez-Acosta²

¹ Centro de Investigación y Desarrollo de Tecnología Digital (CITEDI-IPN),
Avenida Del Parque No.1310, Tijuana, BC, 22510, México

² MIRAL. R&D, 1047 Palm Garden, Imperial Beach, 91932, USA
{uorozco,mgarciav}@citedi.mx, ramacos10@hotmail.com

Resumen El reconocimiento biométrico se basa en la obtención de patrones digitales de uno o más rasgos físicos de la persona, para verificar identidades o identificar individuos mediante algoritmos de procesamiento matemáticos y estadísticos. Las nuevas aplicaciones han creado una demanda de alta confiabilidad en el reconocimiento de personas para ambientes no controlados. En respuesta a esta necesidad, las investigaciones se han dedicado al desarrollo de sistemas de reconocimiento del iris más flexibles, más fiables y sin interacción con los usuarios. Como resultado, muchos algoritmos han sido propuestos y varios se han convertido en partes fundamentales de productos comerciales. Este artículo presenta un análisis de los algoritmos de procesamiento del iris utilizados en cada etapa del sistema de reconocimiento biométrico aplicado al reconocimiento de personas: etapa adquisición de imagen, etapa pre-procesamiento, etapa codificación y etapa comparación de códigos. Este artículo se concluye con las tendencias tecnológicas de cada etapa del sistema.

Palabras clave: reconocimiento del iris, biometría, video-iris, codificación, reconocimiento a distancia.

1. Introducción

Uno de los rasgos biométricos que ha sido objeto de estudio para la autenticación y reconocimiento de personas en los últimos años es el iris, órgano interno del ojo. La autenticación y reconocimiento biométrico, explota el hecho de que ciertas características biológicas son singulares e inalterables. Una tarjeta de identificación o una contraseña escrita pueden ser interceptadas. Sin embargo, la huella digital, el iris, la geometría de la mano y la voz [1], son características imposibles de perder, transferir u olvidar. La idea central de la biometría, es el estudio de rasgos y características del ser humano que no varían con el tiempo para utilizarse como “identificador biométrico”. La aportación de este artículo recae en la revisión de las técnicas más notables en la tecnología del reconocimiento basado en el iris humano. Se pretende brindar una visión completa pero breve de los avances logrados en las diferentes etapas del sistema de reconocimiento del iris, así como futuros desarrollos que pueden aportar a la consolidación de esta tecnología biométrica.

Después de haber descrito el concepto de biometría y la aportación de este artículo,

se examinará el sistema de reconocimiento basado en iris, analizando en la sección 2, las características importantes que ofrece el iris para utilizarlo como identificador biométrico. En la sección 3, se detalla las etapas del sistema de reconocimiento basado en iris. En la sección 4, se presenta una comparación entre las bases de datos existentes las cuales se utilizan para la validación de sistemas de reconocimiento basado en iris. Posteriormente en la sección 5, se describe la importancia del rendimiento en cada una de las etapas del sistema de reconocimiento basado en iris. Finalmente, en la sección 4, se exponen las conclusiones.

2. Iris como identificador biométrico

El iris se localiza detrás de la cornea y del humor acuoso frente del lente. Este consiste en un tramado de tejidos conectivos, fibras, anillos y coloraciones, que al observarse a corta distancia constituyen una huella distintiva de las personas (figura 1). La textura del iris no tiene expresión genética y su morfogénesis es completamente aleatoria [2].

El iris tiene ventajas prácticas adicionales sobre otros identificadores biométricos como son:

- Escaneo, facilidad de registrar la imagen a cierta distancia sin necesidad de contacto físico.
- Codificación, su estructura permite codificar con una densidad de información de 3.4 bits por mm² de tejido.
- Permanencia, el iris comienza a formarse en el tercer mes de la gestación y sus estructuras son completamente creadas en el octavo mes, después de los 2 o 3 años no presenta cambios.
- Unicidad, probabilidad de que dos personas tengan el mismo patrón es cero.
- Fiabilidad, imposibilidad de modificarlo sin riesgo de perder la visión.
- Cuantificación, su facilidad de registro a corta distancia.
- Aceptabilidad, el reconocimiento del iris según los estudios realizados no causa daño al identificado, por lo tiene un buen porcentaje de aceptación.



Fig. 1. Ubicación del iris, pupila y esclera [3],

Sin embargo, el iris presenta algunas desventajas que dificultan su utilización como identificador biométrico [2]:

- Su tamaño pequeño dificulta la adquisición a ciertas distancias.
- Es un objetivo móvil, el movimiento del ojo podría dificultar su captura.
- Está ubicado en una superficie curva, húmeda y reflexiva.
- Su imagen suele estar afectada por pestañas, párpados y reflejos de luz, limitando el área de interés en la cual contiene las características biométricas.

Aunque el iris señale algunas desventajas, en la tabla 1 se muestra un comparativo de los diferentes identificadores biométricos que se pueden utilizar resaltando al iris por su alta fiabilidad, estabilidad y la prevención de ataques. Por ejemplo, una firma la pueden copiar, por lo que su fiabilidad es baja.

Tabla 1. Comparación entre los diferentes Identificadores Biométricos [1].

	Fiabilidad	Facilidad de uso	Prevención de ataques	Aceptación	Estabilidad
Ojo (Iris)	Muy alta	Media	Muy alta	Media	Alta
Ojo (Retina)	Muy alta	Baja	Muy alta	Baja	Alta
Huellas dactilares	Alta	Alta	Alta	Alta	Alta
Geometría mano	Alta	Alta	Alta	Alta	Media
Escritura y firma	Media	Alta	Media	Muy Alta	Baja
Voz	Alta	Alta	Media	Alta	Media
Rostro	Alta	Alta	Media	Muy alta	Media

Por lo mencionado anteriormente, se considera al patrón del iris como un identificador biométrico de los más convenientes. Es por ello el gran interés en el área de investigación para que en un futuro se pueda utilizar en entornos de alta seguridad.

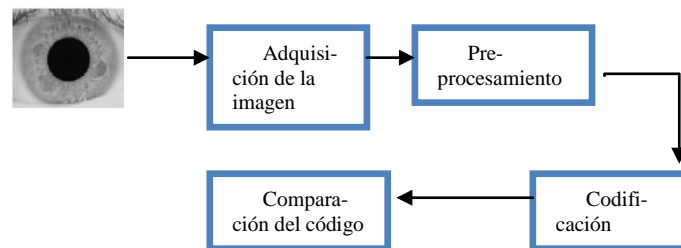


Fig. 2. Sistema de reconocimiento del iris [4].

3. Sistema de reconocimiento basado en iris

Un sistema de reconocimiento del iris consta de cuatro etapas principales:

1. Adquisición de la imagen.

2. Pre-procesamiento de los datos capturados (excluye la información innecesaria y preserva aquella que contenga información útil del iris).
3. Codificación (Extracción de características propias del iris).
4. Comparación de las características extraídas, con el patrón iris previamente almacenado.

Estas cuatro etapas fundamentales se muestran en la figura 2.

3.1. Adquisición de la imagen

La adquisición de la imagen es una etapa trascendental en el reconocimiento; ya que se desea utilizar una imagen con buena calidad dado que puede afectar el rendimiento e interoperabilidad del sistema [5]. La adquisición puede realizarse de forma manual o automática utilizando cámaras convencionales de luz visible o infrarroja [6]. En la forma manual, el usuario debe tener un entrenamiento adecuado, ajustando la cámara para tener el iris enfocado. No obstante en la forma automática se tiene un grupo de cámaras que localizan la cara y el iris haciendo la captura más sencilla. Daugman [7] propuso un sistema de forma manual, donde las imágenes fueron tomadas sin cámaras activas, utilizando espejos y video para ayudar a los usuarios a posicionar sus ojos dentro del campo visual de la cámara. Por otra parte, Wildes et al [8,9] en su sistema utiliza una cámara, para capturar varias imágenes, de las cuales se seleccionaba la de mayor contraste en el límite entre el iris y la esclera.

Entre los sistemas para la adquisición de la imagen de forma automática más sobresalientes esta el sistema auto adaptivo [10] que utiliza una cámara de ángulo amplio (W-Cam), y una de ángulo estrecho (N-Cam). La W-Cam captura la cara y la N-Cam los ojos. Cuando la imagen del ojo aparece en el punto de referencia en la vista de W-Cam la imagen del ojo aparecerá en la vista de N-Cam. Como iluminación se utiliza un arreglo de leds infrarrojos con una emisión de 20 grados que se mueve junto con las cámaras de baja resolución.

Otro sistema automático utiliza un sistema doble de ojo, un espejo integrado con dos cámaras y una pantalla LCD. En este sistema el usuario puede ver sus ojos en el LCD de la misma manera que se ve en el espejo, pero las cámaras reales están debajo del LCD siendo invisible. Para facilitar la captura al usuario hay una luz que se pone verde o roja. Así mismo Wenbo et al [11] propuso un sistema de captura automática de larga distancia. En la cual el usuario pasa por una puerta y se detiene entre 2~3 segundos para ser identificado. Las cámaras tienen alta resolución (2352×1728 píxeles), una velocidad de transferencia de 14 cuadros por segundo y un lente de 300 mm. Permitiendo capturar el iris en un rango de 16mm x 12mm en tiempo real a 3 metros de distancia.

La adquisición de la imagen es una etapa clave del sistema, donde se requiere capturar una imagen de alta calidad, ya que en caso contrario puede afectar al rendimiento del sistema. Por ello se requiere de varios factores de ayuda que guíen a los usuarios a posicionarse correctamente para la adquisición. Sin embargo, estos factores pueden disminuir si se utilizan más cámaras de apoyo o técnicas auto adaptivas para encontrar los rasgos faciales buscando minimizar la interacción del usuario con el sistema.

3.2. Pre-procesamiento

Todas las imágenes obtenidas en la etapa de adquisición contienen además de la información del iris, la región que lo rodea como párpados, pestañas y áreas que forman parte del rostro. Por ello, es necesario realizar un pre-procesamiento de la imagen antes de extraer las características biométricas de la región del iris.

Este pre-procesamiento consta de dos sub-etapas: La *Segmentación* y la *normalización*.

3.2.1. Segmentación

En la etapa de segmentación se requiere aislar la región del iris y excluir al resto de la imagen, primero se localizan los bordes entre iris-pupila e iris-esclera, obteniendo dos círculos donde los parámetros del radio y centro son parámetros importantes a obtener. Una mala segmentación trae consigo la disminución del rendimiento del sistema completo. En la figura 3 se muestra la diferencia entre una segmentación adecuada y una errónea. Para realizar la segmentación de la región del iris, existen varias técnicas como la ecuación integro-diferencial de Daugman [12]. Esta ecuación se utilizó para buscar el borde iris-esclera, de tal modo que detecte los cambios más abruptos iris (oscuro) y esclera (blanco). Después se continúa para obtener el borde entre el iris-pupila.

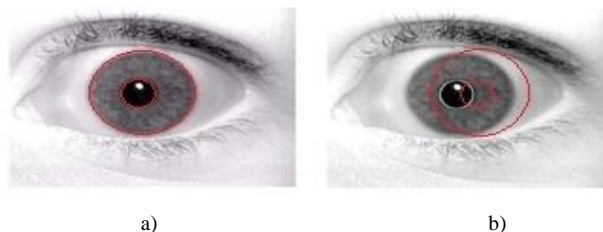


Fig. 3. Segmentación de la región del iris. a) Correcta. b) Incorrecta.

El sistema desarrollado por Wildes et al [8] propone una alternativa para localizar los parámetros de los bordes. En su trabajo realiza una transformación de la imagen a un mapa binario de contornos, fijando un umbral en la magnitud de los gradientes de la imagen y obteniendo su dirección. La información resultante permite considerar sólo los bordes verticales, evitando con esto, que los párpados, pestañas pudieran influenciar en la detección de los bordes.

S. Lim, K. Lee y J. Kim [13] propusieron un modelo de contornos activos para la detección de la pupila y párpados. Éste toma en consideración los límites de la pupila siendo un contorno cercano-circular. Además, utiliza un algoritmo para mejorar la detección del párpado excluyendo información relevante principalmente por encima y debajo del párpado permitiendo una mejor segmentación de la región del iris.

Kong y Zhang [14] proponen un nuevo modelo de detección del ruido para la segmentación exacta de la región del iris. Las pestañas, los párpados y la reflexión son tres ruidos principales. Existen otros métodos que han resuelto la eliminación del párpado. Sin embargo, este modelo detecta y elimina las pestañas así como las reflexiones obteniendo una mejor imagen del iris.

3.2.2 Normalización

En esta etapa se obtiene una nueva imagen de la región del iris. Una vez que el iris ha sido localizado, se genera la nueva imagen para cada iris con cierto tamaño estandarizado y así permitir comparaciones. Para esto, se utiliza el método propuesto por Daugman [12], en donde se mapea la región del iris a una imagen rectangular de dimensiones constantes.

La región anular del iris genera una región rectangular de dimensiones constantes, esto facilita la interpretación y la carga computacional.

En esta etapa de pre-procesamiento se requiere identificar la ubicación del iris para extraer su patrón y eliminar toda información que pudiera propiciar un mal reconocimiento afectando en el rendimiento del sistema. Daugman presenta una aproximación muy cercana de lo que se requiere haciendo sus operaciones en el centro del ojo. En cambio los otros métodos presentados, identifican la ubicación del iris así como la detección de pestañas, párpados y reflexiones para excluirlas de la codificación.

3.3. Codificación

Una vez que se tiene una imagen con información exclusiva de la región del iris, se debe de extraer la información más discriminativa, codificando las características más significativas. Muchos autores utilizan filtros multicanal de Gabor [12,15], Log-Gabor [16], Transformada rápida de Fourier [17] y la Transformada de Wavelet [18]. Los filtros multicanal consiguen extraer los coeficientes que finalmente conformaran la plantilla digital biométrica del iris.

Los filtros de Gabor, proporcionan la representación de una señal en espacio y frecuencia, se crean modulando una onda Gaussiana con una onda sinusoidal. La parte real es especificada por una función coseno modulado con la Gaussiana y la parte imaginaria es especificada por una función seno modulada por una Gaussiana. Daugman utilizó filtros de Gabor bidimensionales para la codificación de la región del iris especificando la posición de la imagen, el largo y ancho efectivo, haciendo una demodulación en la salida de los filtros para comprimir los datos cuantificando la fase para cada cuadrante en el plano complejo. Esto permite desechar la información redundante como la iluminación obteniendo por cada pixel de la imagen dos bits.

Los filtros de Log-Gabor permiten una representación más compacta de imágenes cuando el ancho de banda de las mismas es mayor a una octava. Este parece ser el caso de las imágenes del iris, donde la información de su textura tiene un ancho de banda superior a una octava. Sin embargo, esta componente puede ser eliminada para cualquier ancho de banda usando un filtro Gabor, donde la Gaussiana tenga una escala logarítmica. A esta clase de filtros se les conoce como filtros Log-Gabor.

La transformada rápida de Fourier “FFT” realiza una conversión en espiral logarítmica a intervalos de 50 píxeles, donde los picos que se observaron en las bandas más externas fueron producidos por los párpados.

Por su parte, **la transformada Wavelet** se utiliza para descomponer la información de la región del iris en sus componentes que aparecen para diferentes resoluciones. En comparación con la FFT tiene una mayor compresión. Además ésta detecta las características concretas de las imágenes o la eliminación de ruido asociado a las señales. Para obtener un vector característico se aplica el banco de filtros wavelet con una

dimensión de 87. Cada localidad en el vector tiene un valor entre -1 y 1 pero se cuantifica para que los valores positivos sean representados por “1” y los negativos por un “0”.

Estas técnicas presentan una interesante forma de extraer la información que es de utilidad, Log-Gabor es mejor filtro para codificar la imagen por el ancho de banda que requiere. Por su parte, la FFT presenta una forma sencilla de codificación, no obstante, la transformada wavelets comprime los datos en mayor proporción.

3.3. Comparación de códigos

Después de haber realizado todas las etapas anteriores, la plantilla digital biométrica del iris obtenida, se almacena en una base de datos (BD) para futuras comparaciones. Cuando una identificación o autenticación es requerida, se repite cada una de las etapas aunque en lugar de almacenar la plantilla digital biométrica del iris, ésta es comparada con todas o sólo una plantilla digital biométrica de la BD según sea el requerimiento de autenticación o identificación.

Para lograr una comparación exitosa es necesario definir una serie de métricas de distancias que permitan medir la diferencia, entre las plantillas digitales biométricas del usuario y las almacenadas en la base de datos del sistema. Estas distancias, permiten establecer una medida de desigualdad entre las plantillas digitales biométricas a ser comparadas. Aunque existen varias técnicas, las dos más representativas para obtener la distancia son: Euclidiana [18] y Hamming [7].

La distancia Euclidiana centra la clase en una plantilla digital biométrica de características de un número de muestras inicialmente tomadas. La determinación del umbral indicará la pertenencia a una clase o a otra. Esta medida está dada por el valor de la distancia que marca el límite entre estas clases.

La literatura reporta que **la distancia de Hamming** es comúnmente utilizada para el reconocimiento de patrones. Esta técnica es una medida de la variación entre el valor de la plantilla digital biométrica del iris actual y el valor de la plantilla digital biométrica que está almacenada en la base de datos.

Wildes en su trabajo [8] utilizó la distancia Hamming. Sin embargo, modificó las ecuaciones para compensar la variabilidad producida por la escala, la rotación y los saltos en la imagen. Wildes calculó el nivel de emparejamiento obtenido de la integración de las diferencias de pixel en todas las bandas de frecuencia disponibles.

Es importante determinar la mejor métrica de correlación para decidir si una plantilla digital biométrica corresponde o no a alguna plantilla digital biométrica almacenada en la base de datos. Dado que la comparación se debe realizar a toda la base de datos, esta métrica deberá presentar cálculos muy exactos. Por ello se debe tener un algoritmo eficiente para poder realizar dichas comparaciones de forma casi inmediata.

4. Base de datos

Para almacenar las plantillas digitales biométricas se utiliza una base de datos administrada por un sistema gestor de base de datos, el cual permita almacenar grandes cantidades de plantillas digitales biométricas y posteriormente acceder a ellas de forma rápida y estructurada sin que éste se sature o colapse. Sin embargo, durante el

proceso de prueba de los algoritmos desarrollados para cada una de las etapas del sistema de reconocimiento del iris, se han contrastado los resultados obtenidos en el uso de diferentes bases de datos de iris de carácter público [19]. La tabla 2 muestra un comparativo de las diferentes bases de datos que se utilizan como CASIA Iris Database [20] (China), Palacky University Iris Database [21]. (República Checa), University of Bath Iris Image Database [22] (Inglaterra), GBTNI II [23] (Madrid).

Cada una de las bases de datos tiene alguna característica importante para ser utilizada pero en este artículo se remarca el uso de dos bases de datos. La base de datos de CASIA, es en su gran mayoría de personas asiáticas. Dado que la característica propia de esa población son los ojos rasgados, es un factor que influye de manera notable a la hora del pre-procesamiento de las imágenes del iris. Así mismo, la base de datos de la universidad de BATH maneja un número grande de muestras poniendo a prueba el rendimiento del sistema.

Tabla 2. Comparación de distintas Bases de Datos de Iris.

	CASIA	PALACKY	BATH	GBTNIv2
Sesiones	2	1	20	2
personas	80	64	200	--
Ojos diferentes	108	128	4000	
Muestras	756	584	8000	
Formato	BMP	PNG	J2C	BMP
Resolución	320x 280	576 x 768	1280x 960	1392x 1040
Color	Escala de Grises	Color	Escala de Grises	Escala de Grises
Iluminación	Infrarroja	--	Infrarroja	Infrarroja
Máscara	No	Si	No	No

5. Rendimiento del sistema

En un sistema de identificación de personas se requiere que el reconocimiento proporcione un resultado veraz y en el menor tiempo de ejecución posible. Cada etapa del sistema contribuye consumiendo un instante de tiempo. En la tabla 3 se muestran los tiempos de ejecución de cada etapa en el sistema de Daugman [7].

En el sistema de Daugman [7] se utiliza un procesador a 300-MHz RISC, realizando 100,000 comparaciones de diferente iris por segundo o un millón de comparaciones en un servidor de 1.7 a 2 GHz.

El número de comparaciones y la velocidad de ejecución de cada etapa del sistema son afectadas por las diferentes arquitecturas de hardware como el desarrollo de software. En la actualidad, el poder de cómputo ha incrementado y su precio de adquisición ha disminuido. Aunado a ello, las técnicas de programación han mejorado dando la oportunidad a utilizar computadoras personales como procesadores de uso específico disminuyendo tiempos de procesamiento. En el futuro se tiene previsto que los

identificadores biométricos sean clave en sistemas de acceso y de seguridad nacional, dado el incremento en las exigencias respecto a fiabilidad, alto rendimiento y eficacia.

Tabla 3. Tiempos de Ejecución de las Etapas del Reconocimiento del Iris en el Sistema de Daugman.

Etapas	Tiem- po
Calcular el foco de imagen	15 mseg
Quitar reflexiones espectaculares	56 mseg
Localizar ojo e iris	90 mseg
Límite de la pupila	12 mseg
Detectar y limitar los dos párpados	93 mseg
Remover pestañas y forma de lentes de contacto	78 mseg
Demodulación y creación del iriscode	78 mseg
Comparación XOR de los dos iriscode	12µseg

6. Conclusiones

En la actualidad se pueden utilizar diferentes formas de identificación, como claves, dispositivos electrónicos entre otras cosas. Sin embargo, es conveniente contar con un identificador único, inalterable, difícil de extraviar. Las propiedades físicas del iris tienen grandes ventajas para ser utilizado como método de identificación. Por ello, en este artículo se revisó todas las etapas del sistema de reconocimiento basado en iris. Se describió que para mejorar el rendimiento del sistema se requiere de la adquisición de una imagen de muy buena calidad. Con respecto a la etapa de Segmentación, la mayoría de los trabajos de investigación se basan en el trabajo de Daugman y de su ecuación integro-diferencial. Así mismo, para realizar la normalización de la imagen se emplea la técnica de Daugman, en donde se mapea la región del iris a una imagen rectangular de dimensiones constantes. Para extraer las características del iris se utiliza los filtros de Log-Gabor. En la etapa de comparación, se emplea la distancia de Hamming con mayor frecuencia. Dentro de las bases de datos públicas para analizar los sistemas, la base de datos de CASIA es una de las más referenciadas.

El reconocimiento del iris es un campo amplio de estudio donde la literatura indica que existen muchas técnicas para incrementar su eficiencia. Sin embargo, en la práctica todavía no se tiene la confianza plena de su uso; dado que varios factores pueden alterar las etapas del sistema reduciendo su rendimiento. Por ello se requiere continuar investigando para consolidar al sistema del reconocimiento basado en iris, como uno de los métodos biométricos más confiables para la autenticación e identificación de personas.

Agradecimiento: Esta investigación fue auspiciada por el proyecto SIP2012 del IPN.

Referencias

1. Sanderson S., Erbetta J.: Authentication for secure environments based on iris scanning technology. IEE Colloquium on Visual Biometrics (2000)
2. Muron A., Pospisil J.: The human iris structure and its usages. *Physica*, 39:89-95. Czech. (2000)
3. Iris.: <http://www.cse.msu.edu/~chenyi1/Research/iris/segmentation/iris.jpg> (2012)
4. Terissi L.D., Cipollone L., Baldino P.: Sistema de reconocimiento de iris. FCEIA, Universidad Nacional de Rosario, Argentina (2000)
5. Kalka N.D., Dorairaj V., Shah Y.N., Schmid N.A., Cukic B.: Image Quality Assessment for Iris Biometric. Lane Department of Computer Science and Electrical Engineering West Virginia University, Morgantown, WV-26506, US (2009)
6. Khaw P.: Iris recognition technology for improved authentication. SANS Institute (2002)
7. Daugman J.: How iris recognition works. *IEEE Trans. Circ. Syst. Video Tech.* 14(1), 21-30 (2004)
8. Wildes R.P., Asmuth J.C., Green G.L.: A System for Automated Recognition, 0-8186-6410-X/94, IEEE (1994)
9. Wildes R.P.: Iris recognition: An Emerging Biometric Technology. *Proceedings of the IEEE*, vol. 85, no. 9 (1997)
10. Dong W., Sun Z., Tan T., Qiu X.: Self-adaptive iris image acquisition system. National Laboratory of Pattern Recognition, Institute of Automation, Academy of Sciences, No.95 Zhongguancun East Road, Beijing, China (2008)
11. W. Dong., Z. Sun., T. Tan.: How to Make Iris Recognition Easier. *Proc. Int'l Conf. Pattern Recognition*, pp. 1-4, (2008)
12. Daugman J.: High confidence visual recognition of persons by a test of statistical independence. *IEEE Transactions On Pattern analysis and Machine Intelligence*, vol.15, no 11 (1993)
13. Lim S., Lee K., Kim J.: Efficient iris recognition by characterizing key local variations. *IEEE Transactions On Image Processing*, vol. 13, no. 6 (2004)
14. Kong W., Zhang D.: Accurate iris segmentation based on novel reflection and eyelash detection model. *Proceedings of 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing*, Hong Kong (2001)
15. Cruzm L.F., Carranza F.: Reconocimiento del Iris. Universidad Nacional de Trujillo Perú (2006)
16. Field D.: Relations between the statistics of natural images and the response properties of cortical cells. *Journal of the Optical Society of America*, vol. 4, no. 12, pp. 2379–2394 (1987)
17. Greco J., Kallenborn D., Enchiva M.: Statistical pattern recognition of the iris (2003)
18. Boles W., Boashash B.: A human identification technique using images of the iris and wavelet transform. *IEEE Transactions on Signal Processing*, vol .45, no 4 (1998)
19. Coomonte R.: Sistema de reconocimiento de personas mediante su patrón de iris basado en la transformada wavelet. Proyecto Fin de Carrera publicado en Mayo (2006)
20. CASIA.: CASIA Iris Image Databases, Center for Biometrics and Security Research, Institute of Automation Chinese Academy of Sciences. P.R. China (2012)
21. Michal Dobeš., Libor Machala.: Iris Database, Dept. Computer Science, Palacky University in Olomouc (2012)
22. University of Bath.: Iris Image Database. <http://www.smartsensors.co.uk/information/bath-iris-image-database/> (2012)
23. Grupo de Biometría, Bioseñales y Seguridad (GB2S):. base de datos GBTNI II <http://sites.google.com/site/upmgb2s/bases-de-datos/iris-cara/caracteristicas> (2012)

Sistemas inteligentes de tutoría y aprendizaje

Sistema tutor inteligente para la enseñanza de la programación orientada a objetos con el lenguaje C#

Roberto J. Estrada Benavides, Laura Villavicencio Gómez y Boris A. Aranda Benítez

Universidad Tecnológica de la Región Norte de Guerrero (UTRNG),
Instituto Tecnológico de Zacatepec (ITZ),
Calzada Tecnológico no. 27 col. Centro, Zacatepec, Morelos, México
baaranda@yahoo.es

Resumen Con la finalidad de proporcionar a los docentes de la carrera de Tecnologías de la Información y Comunicación de la Universidad Tecnológica de la Región Norte de Guerrero una herramienta de apoyo en el proceso de enseñanza-aprendizaje se plantea la creación de una herramienta de software que sea capaz de dar tutoría sobre algún contenido temático de las materias que conforman el plan de estudios, de la misma manera en que lo realiza un especialista de la materia, proporcionando una serie de objetos de aprendizaje que permita a los alumnos lograr un aprendizaje significativo sobre la técnica de la Programación Orientada a Objetos. Se hace uso de las nuevas tecnologías de acceso a datos y herramientas de modelado de entidades además de la estructura de los sistemas tutores inteligentes para hacer posible su creación e implementación en los equipos de cómputo de los laboratorios del campus de la UTRNG ubicada en la ciudad de Iguala de la Independencia Guerrero.

Palabras clave: Sistema Tutor Inteligente, Inteligencia Artificial, Enseñanza-aprendizaje.

1. Introducción

El avance que han sufrido las tecnologías de la información y comunicación (TIC) en los últimos años, ha impactado la educación y plantea nuevos requerimientos en los planes y programas de estudio en general sobre todo en los procesos de enseñanza en particular. Dentro del sector educativo el uso de las TIC ha cambiado la forma en que se desarrolla el proceso de Enseñanza-Aprendizaje, el uso apropiado de las TIC en la educación puede facilitar el aprendizaje de cualquiera de las materias y desarrollo de habilidades en los estudiantes. Asimismo desde un punto de vista específicamente instructivo, las experiencias de enseñanza desarrollada con las TIC han demostrado ser más motivantes para los alumnos y eficaces en el logro de ciertos aprendizajes comparada con los procesos tradicionales de enseñanza, basados en la tecnología impresa.

En su continuo esfuerzo por mejorar la calidad de sus alumnos la UTRNG busca implementar diversos mecanismos que le permitan entregar a las empresas profesionistas mejor preparados, una de estas alternativas es la implementación a corto plazo de los Sistemas Tutores Inteligentes (STI's) para apoyar en el proceso de aprendizaje de las materias que componen la currícula de las carreras que ofrece la

UTRNG, buscando en un futuro no muy lejano poder ofrecer a la población estudiantil del país el estudio de sus diferentes carreras vía Web.

2. Sistemas tutores inteligentes (STI)

El propósito del STI es presentar un comportamiento similar al de un tutor humano, que se adapte a las necesidades del estudiante, identificando la forma en que el mismo resuelve un problema para poder brindarle ayuda cuando cometa errores. Las interacciones entre el alumno y el docente, están enmarcadas en las teorías de aprendizaje y de enseñanza aplicables a dicha interacción, lo que brinda un marco teórico al problema citado, donde no solo es importante el conocimiento que debe ser facilitado por el docente hacia el alumno, sino que cobra importancia la forma en la que este conocimiento es presentado, ya que se pretende mejorar el proceso de adquisición y construcción de conocimiento.

Un Tutor Inteligente, por lo tanto: “es un sistema de software que utiliza técnicas de Inteligencia Artificial (IA) para representar el conocimiento e interactúa con los estudiantes para enseñárselo” (Wenger, 1987). Wolf (1984) define los STI como: “sistemas que modelan la enseñanza, el aprendizaje, la comunicación y el dominio del conocimiento del especialista y el entendimiento del estudiante sobre ese dominio”. Giraffa (1997) los delimita como: “un sistema que incorpora técnicas de IA a fin de crear un ambiente que considere los diversos estilos cognitivos de los alumnos que utilizan el programa”.

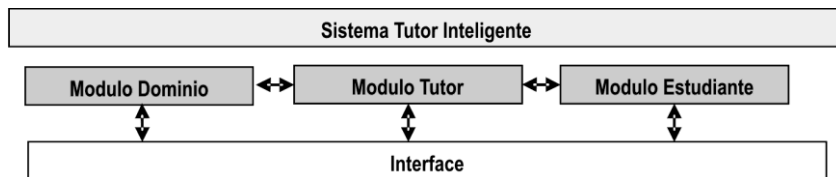


Fig. 1. Estructura de un STI por Carbonell (1970).

La estructura general de un STI ver Fig.1 con la división de los submódulos en funciones específicas soporta una configuración distribuida.

El **Modulo Tutor** del STI define y aplica una estrategia pedagógica de enseñanza, contiene los objetivos a ser alcanzados y los planes utilizados para alcanzarlos. Selecciona los problemas, monitorea el desempeño, provee asistencia y selecciona el material de aprendizaje para el estudiante. Integra el conocimiento acerca del método de enseñanza, las técnicas didácticas y del dominio a ser enseñado.

El **Módulo Estudiante** del STI tiene por objetivo realizar el diagnóstico cognitivo del alumno, y el modelado del mismo para una adecuada retroalimentación del sistema. Se han planteado para el Módulo estudiante los siguientes submódulos (ambos almacenan los datos en una Base de Datos del Estudiante, con un gestor para la misma).

- **Estilos de aprendizaje:** compuesto por una base de datos con los estilos de aprendizajes disponibles en el sistema.

- **Estado de conocimientos:** contiene el mapa de conocimientos obtenido inicialmente a partir del módulo del dominio y que progresivamente el actualizador de conocimientos irá modificando a través de los resultados obtenidos en las evaluaciones.
- El **Módulo Dominio** tiene el objetivo global de almacenar todos los conocimientos dependientes e independientes del campo de aplicación del STI. Básicamente deberá tener los submódulos siguientes:
 - Parámetros Básicos del Sistema: los cuales se almacenan en una base de datos.
 - Conocimientos: son los contenidos que deben cargarse en el sistema, a través de los conceptos, las preguntas, los ejercicios, los problemas y las relaciones.
 - Elementos Didácticos: Son las imágenes, videos, sonidos, es decir material multimedia que se requiere para facilitarle al alumno apropiarse de conocimiento en la sesión pedagógica.

3. Planteamiento del problema

Dentro de la UTRNG el aprendizaje de la Programación de Computadoras y mas aún la Programación Orientada a Objetos no ha logrado ser lo suficientemente significativo para los estudiantes debido a diversos factores tales como la pasividad de los alumnos, el poco tiempo que se les proporciona dentro de los laboratorios, catedráticos que no cuentan con la experiencia en esta técnica de programación entre otras causas, esto se ha reflejado en un alto índice de reprobación que se ha registrado al impartir esta materia, además de que los estudiantes que logran acreditar tienen deficiencias marcadas al aplicar esta técnica de programación a los problemas de la vida real.

Los tutores humanos deben ser expertos en dos aspectos: en el conocimiento técnico y en el área de tutorías y tiene un rol compuesto de tres aspectos fundamentales: diagnostican los problemas del estudiante, planean la retroalimentación y se comunican con los estudiantes, con base a lo anterior las preguntas que surgen son: ¿Cómo logra un tutor humano un aprendizaje realmente significativo en sus estudiantes? ¿Qué métodos o técnicas permiten una participación activa del alumno? ¿Cómo minimizar el índice de reprobación en las materias de Programación? ¿Existe una forma de mejorar el aprendizaje de la Programación Orientada a Objetos?

4. Objetivo general

Proporcionar una herramienta de apoyo a los catedráticos y a los estudiantes de la Universidad Tecnológica de la Región Norte de Guerrero para la enseñanza de la Programación Orientada a Objetos mediante el desarrollo de un Sistema Tutor Inteligente basado en Objetos de Aprendizaje.

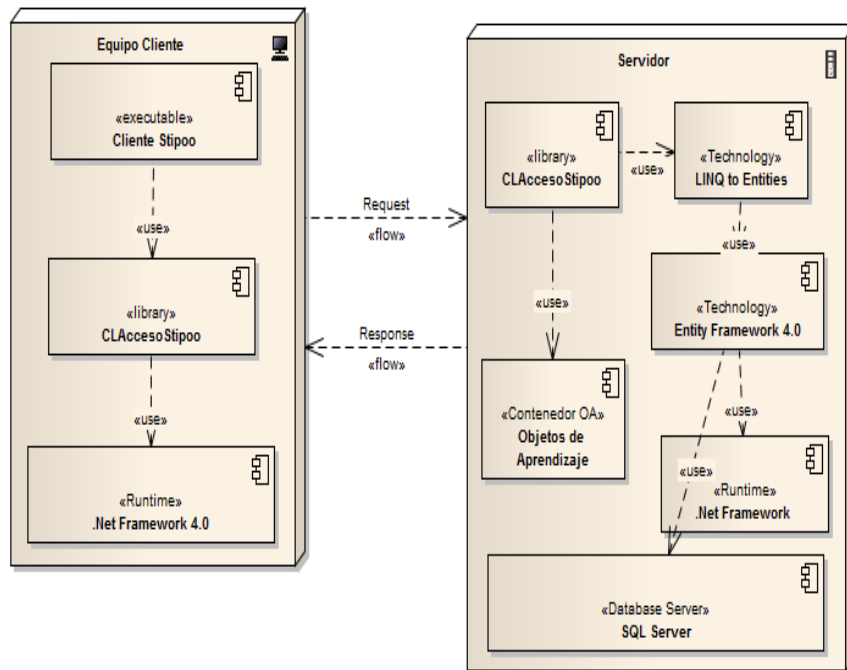


Fig. 2 Diseño General del STI.

5. Objetivos específicos

- Brindar a la comunidad estudiantil de la UTRNG un especialista en Programación Orientada Objetos en el día y el horario que alumno lo requiera.
- Facilitar la asesoría a los alumnos.
- Diagnosticar aspectos en los que algún alumno requiera apoyo y brindarle el material necesario.
- Disminuir el índice de reprobación en la asignatura de Programación Avanzada mediante el uso de esta herramienta.
- Incrementar el número de alumnos asesorados en esta materia.
- Lograr que el sistema brinde la retroalimentación más adecuada para el alumno basado en el diagnóstico realizado, brindándole los objetos de aprendizaje que faciliten la comprensión del tema.

6. Diseño general del STI

El Sistema Tutor Inteligente para la Programación Orientada a Objetos se ejecuta en un esquema Cliente – Servidor y la comunicación se realiza a través de formularios

que invocan librerías *.dll* que a su vez utilizan **LINQ to Entities** para el acceso a datos, los elementos que componen el sistema y su funcionalidad se muestran en la figura 2 y se describen a continuación:

1. **Programa Cliente ejecutable (.EXE)** es el encargado de interactuar con los usuarios del Sistema Tutor Inteligente recibe las peticiones de éste y las envía a la librería de clases encargada del acceso a datos.
2. **Librería de Clases CLAcceso (.DLL)** representa la capa controladora y la de acceso a datos. Éste contiene toda la lógica necesaria para llevar las peticiones que el usuario envió a la capa de vista para procesarla en la capa de modelo una vez hecha las operaciones solicitadas envía de regreso los resultados hacia la capa de vista para que puedan ser presentados al usuario.
3. **.Net Framework** es un marco de trabajo que hace énfasis en la transparencia de redes, con independencia en la plataforma de hardware que permita un rápido desarrollo de aplicaciones, su meta es lograr un acceso simple y universal a todo tipo de información desde cualquier dispositivo, cabe mencionar que en las versiones más recientes de los sistemas operativos de Microsoft el .Net Framework viene como parte del sistema.
4. **LINQ** es un lenguaje integrado de consultas que trabaja principalmente sobre objetos o entidades, su principal objetivo es unificar la forma de acceder a distintos orígenes de datos tales como Bases de Datos, archivos de Texto, archivos XML, Servicios WEB ya que es un lenguaje de consultas que utiliza la sintaxis del lenguaje C# (y en cualquier lenguaje contenido en la suite de visual studio .NET) esto permite una mayor productividad debido a que no se invierte tiempo en aprender lenguajes nuevos como SQL, xquery para acceder a datos.
5. **Entity Framework** es una tecnología de Microsoft que nos permite mapear tablas de bases de datos de distintos proveedores (MySQL, SQL Server, Microsoft Access, Oracle) a un modelo de entidades para poder trabajar con ellos como objetos, e inclusive se puede generar el modelo de entidades (clases) y a partir de éste generar la base de datos.
6. **Contenedor de Objetos de Aprendizaje** es una ubicación dentro del servidor que contiene de manera física los distintos archivos que servirán como herramienta de aprendizaje para que el alumno logre un aprendizaje significativo de los temas que componen la materia en cuestión.
7. **SQL Server** es el servidor de base de datos que almacenará y administrará toda la información referente al Sistema Tutor Inteligente.

7. Resultados del sistema tutor inteligente para la enseñanza de la programación orientada a objetos (STIPOO)

El sistema se inicia al lanzar el archivo ejecutable de la aplicación en el equipo cliente, el cual muestra al usuario el formulario principal que muestra un botón de inicio que contiene las opciones disponibles, la barra de estado del formulario principal despliega la información del usuario que se encuentra en sesión actualmente, tiene además los botones de maximizar, minimizar y cerrar la ventana de la aplicación. Ver figura 3.

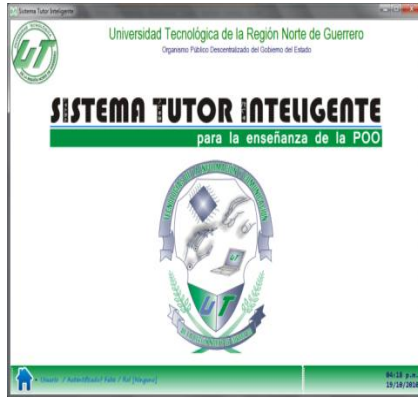


Fig. 3 Pantalla de Bienvenida del STiPoo



Fig. 4 Crear un Nuevo Curso en el STI.

Cabe mencionar que el Sistema Tutor Inteligente utiliza un esquema de seguridad basada en roles de usuario, existen 3 diferentes tipos de usuarios que utilizarán el sistema, entre los cuales tenemos:

1. **Administrador**.- El cual tiene permiso de acceder a cualquiera de las opciones de los menús, entre sus funciones se encuentran principalmente la creación de cursos, así como estructurar los contenidos temáticos del mismo para que puedan ser utilizados por los demás usuarios del sistema.
2. **Docente**.- Estos usuarios tienen privilegios únicamente sobre la parte académica del sistema, es decir, una vez que un curso es creado y estructurado por el administrador, los docentes pueden comenzar a aportar los objetos de aprendizaje de cada uno de los temas que componen el curso, además son los responsables de crear el banco de preguntas que serán utilizadas en las evaluaciones de los alumnos.
3. **Alumno**.- Son los usuarios finales del sistema su función es acceder a la información de los cursos disponibles en su carrera, una vez realizado esto pueden inscribirse en el curso y comenzar a utilizar la herramienta, el sistema lleva el control de los avances del usuario y cuando éste considera estar listo puede realizar la evaluación de la unidad, hasta que al final acredite exitosamente el curso.

7.1. Crear cursos nuevos al STI (Usuario Administrador)

Una vez que el administrador seleccionó la opción de Agregar Nuevo Curso se le muestra el formulario de la figura 4 para que comience con la captura de la información, es importante mencionar que para poder dar de alta un nuevo curso debe ser parte de alguno de los temarios oficiales que proporciona la Coordinación General de Universidades Tecnológicas, ya que como se mencionó anteriormente alguna de la información sensible para el STI proviene de la base de datos del SAIUT, la captura comienza seleccionando la carrera que es propietaria de la materia de la cual quiere crearse el curso.

7.2. Agregar contenido temático a un curso del STI (Usuario Administrador)

Después de terminar la captura de las unidades del curso se procede a finalizar la estructura del curso agregando los temas que corresponden a cada una de las unidades temáticas, con esta acción se completa el árbol de conocimiento del curso que se ha creado y se encuentra listo para que los docentes y especialistas del conocimiento lo comiencen a alimentar con preguntas referentes a los temas además de que los docentes aporten los objetos de aprendizaje que servirán de herramienta de apoyo a los profesores para el proceso de enseñanza-aprendizaje de cada uno de los cursos

El usuario selecciona el nivel de estudios del curso, después el STI lista los cursos que ya se han creado, posteriormente el usuario selecciona las unidades temáticas del curso seleccionado, al hacer esto el sistema muestra la estructura del curso organizado desde el nombre del curso, las unidades que lo componen y los temas que se han capturado de cada unidad, esta información facilita al usuario la información que necesita agregar, el usuario selecciona la unidad temática, después proporciona el número del tema seleccionándolo, agrega el nombre del tema, así como los puntos que se busca desarrollar en el alumno, dentro del modelo de universidades tecnológicas se tienen tres aspectos fundamentales que son el ser, saber y el saber hacer, los cuales deben especificarse en la captura. Ver Figura 5.

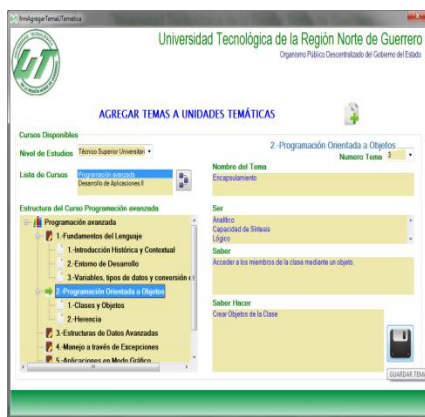


Fig. 5 Árbol de conocimiento del curso



Fig. 6 Agregando un Objeto de Aprendizaje al STI

7.3. Agregar objetos de aprendizaje al STI (Usuario Docente)

Este proceso es de vital importancia debido a que los docentes que son los especialistas del conocimiento aportan estos materiales que van desde documentos, imágenes, archivos de audio, ebooks o videos que servirán de herramienta de apoyo para lograr la comprensión total del tema que se ve en clases, los docentes clasifican los objetos de aprendizaje de acuerdo al tema de la unidad que abarcan y lo registran en el servidor para que el STI pueda disponer de la información en el momento en que un alumno necesite tutoría en algún tema de la materia, en la fig. 6 se ilustra el formulario de captura para agregar un Objeto de Aprendizaje.

Una vez que se introdujeron todos los datos del Objeto de Aprendizaje como el tipo de archivo, el nombre, el autor, el idioma, la fecha de publicación y una breve descripción del material aportado, éste se encuentra listo para ser registrado en la base de datos del STI y también el objeto de aprendizaje será copiado en el servidor para que se encuentre disponible para los alumnos que lo soliciten como se muestra en la figura 7.



Fig. 7 Progreso de copia de un OA



Fig. 8 Agregando preguntas para las evaluaciones

7.4. Agregar preguntas a la base de datos del STI (Usuario Docente)

Otra de las tareas fundamentales de los docentes consiste en agregar preguntas para que puedan ser tomadas en cuenta al crear una evaluación de una unidad temática de algún curso, el esquema de colaboración del STI permite que todos los docentes puedan subir sus preguntas al servidor ya que se considera que la aportación de cada docente enriquecerá el contenido de las distintas evaluaciones que se creen para cada una de las unidades temáticas de los cursos, por otro lado es importante mencionar que por el momento se pueden agregar dos tipos de preguntas al sistema que son de las más utilizadas en todos los cuestionarios de evaluación.

7.5. Crear cuestionario de evaluación al STI (Usuario Docente)

Quizá la tarea más importante del docente sea la de crear un cuestionario de evaluación de una unidad temática de un curso para que sea publicada y pueda estar disponible para evaluar el conocimiento de los alumnos inscritos en él, como se mencionó anteriormente las evaluaciones se componen de varias de las preguntas que todos los docentes de la carrera aportan, los cuestionarios de evaluación se componen de “n” número de preguntas, además se pueden establecer parámetros como cuál va a ser la calificación mínima aprobatoria, la calificación máxima que se puede alcanzar en la evaluación, entre otros valores.

7.6. Inscribirse a un curso disponible en el STI (Usuario Alumno)

Esta tarea es muy importante para los alumnos ya que al llevarla a cabo se realiza un registro en la base de datos que servirá para llevar el control del avance del alumno en el curso al que se registre, la tarea es muy sencilla, al acceder a esta opción se desplegará un formulario en el que se puede verificar cuales son los cursos que se encuentran publicados por el administrador del sistema por lo que solamente es necesario seleccionar el curso al que el alumno desee integrarse para llevar a cabo el registro ver fig. 9.

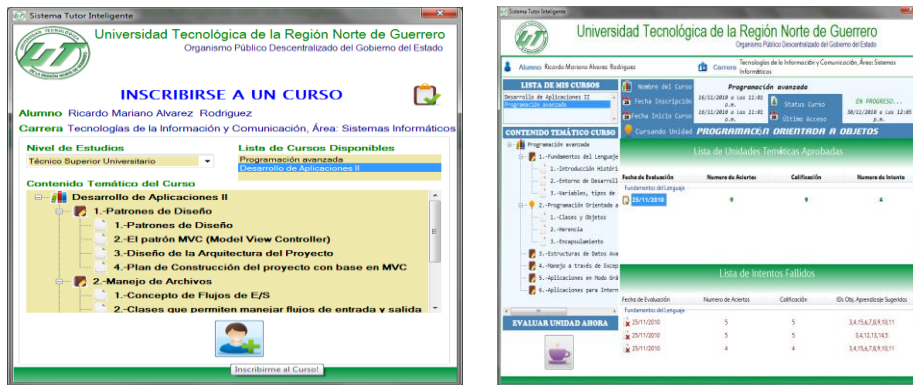


Fig.9 Inscribiendo un alumno a un curso del STI. Fig. 10 Avance de un alumno en un curso

7.7. Verificar avances de un curso en el STI (Usuario Alumno)

En la figura 10 se muestra el formulario donde se observan más datos acerca de los avances del alumno en el curso, aparece información acerca de la fecha en que el alumno inició el curso, así como la fecha en que el usuario acceso por última ocasión al sistema, esta información resulta interesante porque como docentes se puede verificar el tiempo que tarda el alumno en terminar su curso, por otra parte se muestra también el nombre de la unidad temática del curso en el que el alumno se encuentra actualmente, otra información importante para los alumnos es que el STI le muestra al alumno el contenido temático del curso lo que le permite verificar fácilmente cuantas unidades y cuales temas le hace falta cursar para poder aprobar el curso.

7.8. Tomar un examen de evaluación (Usuario Alumno)

En el formulario de la evaluación se tienen datos como el nombre del alumno y la carrera a la que pertenece así como el nombre del curso en el que se encuentra inscrito y el nombre de la unidad temática a evaluar, en este instante todavía puede cancelar el intento de examen cerrando la ventana de la aplicación.

En la parte inferior del formulario se observan los controles de navegación entre las preguntas del cuestionario de evaluación para poder ir a la siguiente pregunta o regresarse a la pregunta anterior, se muestra también un cronómetro que muestra al alumno el tiempo que le tomará completar el examen, sin embargo todavía todos estos

controles se encuentran deshabilitados hasta que el usuario una vez más confirme dando clic en el botón para dar inicio al examen, una vez que el alumno confirma el inicio del examen de evaluación el formulario habilita los controles de navegación y despliega la información de la primer pregunta del cuestionario, la figura 11 ilustra lo anteriormente descrito.



Fig. 11 Evaluación en curso de un alumno



Fig. 12 Resultado de evaluación Aprobatoria.

7.9. Publicando resultados

Después de realizar la evaluación, el usuario termina su examen y lo envía para que el STI lo revise y emita el resultado correspondiente para mostrárselo al alumno, en el primer caso cuando el resultado de la evaluación no es favorable se muestra el resultado en un formulario que detalla el desempeño obtenido por el alumno en cada uno de los temas que componen la unidad temática, en la figura 12 se ejemplifica el resultado obtenido.

8. Conclusiones

El presente trabajo muestra como diseñar e implementar un Sistema Tutor Inteligente para la enseñanza de la programación orientada a objetos el cual emplea objetos de aprendizaje que son aportados por los especialistas de la materia pertenecientes a la carrera de tecnologías de la información y comunicación de la Universidad Tecnológica de la Región Norte de Guerrero. Es importante recalcar que dentro del esquema de las universidades tecnológicas es muy importante la figura del profesor – tutor ya que esta instituido a nivel nacional que por cada 30 alumnos existe un tutor de tiempo completo el cual es el encargado de apoyarlos tanto en lo académico como en lo personal por lo que es de vital importancia contar con una herramienta de apoyo capaz de diagnosticar el conocimiento de la misma forma en que lo hace el especialista de la materia, esto ha permitido atender de forma eficaz a una cantidad mayor de la que normalmente realiza el docente de manera individual, además con la creación de esta herramienta de software los alumnos cuentan con un especialista a la hora en que lo deseen y no están atendidos al horario en que el docente

se encuentre libre, ya que el sistema corre en la red de la institución y se encuentra disponible en cualquiera de los equipos de los laboratorios del campus.

Por otra parte se observó que al involucrar a todos los docentes de la carrera se logró enriquecer de gran manera el contenido de los cuestionarios de evaluación de cada una de las unidades del curso, puesto que al tener diferentes docentes también existe distintas formas de interpretar los contenidos y esto hace que al elaborar sus preguntas y almacenarlas en el STI las evaluaciones sean más completas y por lo tanto se tiene la certeza de que los resultados que arroja el sistema tutor inteligente son confiables y reflejan el nivel de aprendizaje obtenido por cada uno de los alumnos.

Referencias

1. [Carbonell, 1970] Vigotsky, Lev S.: Mind in Society – The development of higher Psychological Processes. Cambridge Harvard University Press (1970)
2. [Wenger, 1987] Artificial Intelligence and Tutoring Systems – Ed. Morgan Kaufman Publishers Inc. (1987)
3. [Troy Magennis 2010] LINQ to OBJECTS USING C# 4.0 - Using and extending Linq to Objects and parallel Linq (plinq). Ed. Addison – Wesley (2010)
4. [Scott Klein 2010] PRO ENTITY FRAMEWORK 4.0 - The future of data access in .NET framework. Ed. Apress (2010)
5. [Christian Nagel, Bill Evjen, Jay Glynn 2010] PROFESSIONAL C# 4 and .NET 4 –Wrox Programmer to Programmer. Ed. Wiley Publishing Inc. (2010)

Un tutor inteligente y afectivo dentro de una red social para aprendizaje de matemáticas

María L. Barrón-Estrada, Ramón Zatarain-Cabada, J.M. de Jesús López Arizmendi, José Luis Olivares Camacho y Karen Yadira González Trapero

Instituto Tecnológico de Culiacán, Juan de Dios Bátiz s/n, Col. Guadalupe, Culiacán Sinaloa, 80220, México
{rzatarain, lbarron}@itculiacan.edu.mx, {manuel.inmortal,joseluokeolivares}@gmail.com

Resumen. En este artículo, presentamos un tutor inteligente y afectivo dentro de una red social para el Aprendizaje de Matemáticas. El tutor inteligente toma en cuenta aspectos cognitivos como afectivos y usa lógica difusa para calcular los ejercicios que le son presentados al estudiante. Por otra parte dentro del trabajo de investigación, se están usando redes neuronales de Kohonen para reconocer emociones por medio de caras y voz. La red social y el sistema tutorial inteligente están integrados en una aplicación web. Los resultados preliminares nos muestran los beneficios de usar un sistema tutorial junto con una red social.

Palabras clave: Sistema tutorial inteligente, computación afectiva, red social, afecto del estudiante.

1. Introducción

El trabajo de un tutor humano es enseñar y capacitar a un estudiante a través de una instrucción individualizada. Esta individualización que el tutor humano realiza con el estudiante es hecha a través de diferentes maneras para adaptar sus materiales educacionales hacia las necesidades de los estudiantes. El objetivo de un Sistema Tutorial Inteligente (STI) no es diferente: ofrecer servicios de enseñanza que apoyen un aprendizaje personalizado. Desde sus orígenes en 1970, la primera generación de sistemas tutoriales inteligentes estaba enfocada a trabajar más con aspectos cognitivos del estudiante [1, 2, 3, 4,5].

Sin embargo, si queremos sistemas computacionales (incluyendo un STI) que sean inteligentes e interactúen con nosotros, debemos otorgarles la habilidad de reconocer emociones [6] y reaccionar a ellas.

La computación afectiva es uno de las áreas más prometedoras en los diferentes campos del aprendizaje y hoy en día es uno de los tópicos de más interés en las más importantes conferencias sobre tecnologías del aprendizaje como “Intelligent Tutoring Systems Conference (ITS)”, “Artificial Intelligence in Education (AIED)”, y “International Conference in Advanced Learning Technologies (ICALT)”. Este tema atrae investigadores de diversos campos incluyendo ciencias computacionales, inteligencia artificial, psicología y educación; en años mas recientes hemos visto un

número creciente de sistemas tutoriales inteligentes afectivos desarrollados para diferentes campos del aprendizaje [7, 8 ,9].

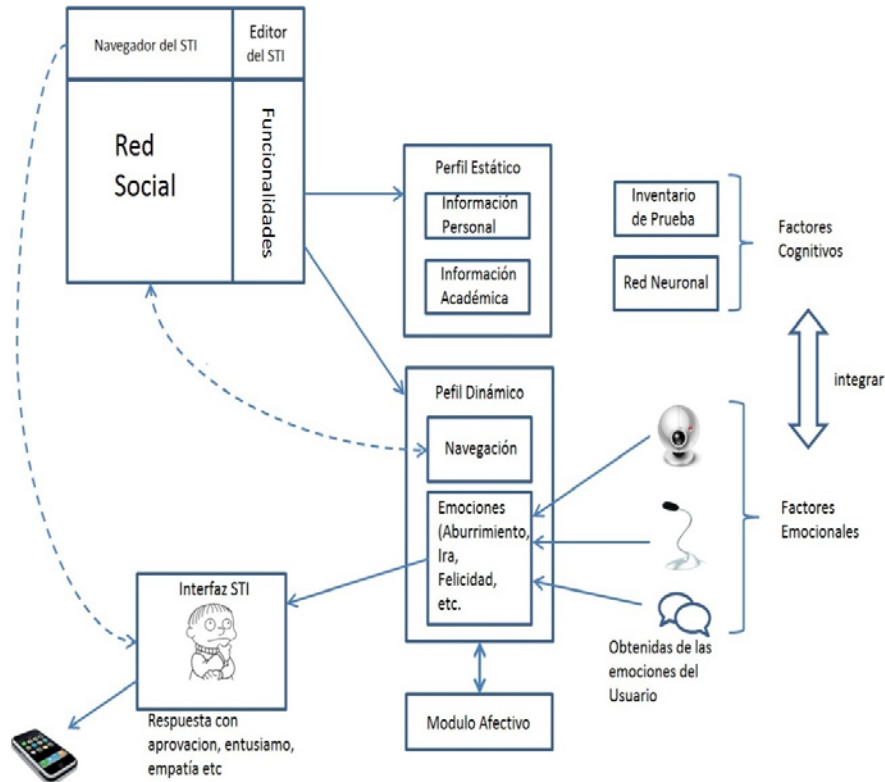


Fig. 1. Red Social para Sistemas de Tutores Inteligentes y Afectivos.

La investigación en computación afectiva incluye detectar y responder a emociones. Los sistemas de detección de emociones observan y estudian el rostro, el habla, la conversación y otros rasgos humanos para detectar frustración, interés, aburrimiento, etc. usando dispositivos con sensores como cámaras web o de lentes azules [10,11], micrófonos [12] y métodos como diálogos conversacionales [13]. Por otro lado, los sistemas de reconocimiento de emociones buscan la mejor manera para manejar las emociones negativas del estudiante. Existen trabajos de investigación excelentes relacionados con este problema [7, 9, 14].

En este artículo presentamos un sistema tutorial inteligente incorporado en una red social de aprendizaje el cual pretende ser utilizado para mejorar los resultados en la prueba ENLACE (Evaluación Nacional del Logro Académico en Escuelas de México). La prueba ENLACE es una evaluación estándar del sistema de educación nacional, aplicado a los estudiantes de primaria y secundaria. Esta prueba mide el nivel de aprendizaje en matemáticas, español y una tercera materia que cambia cada año. Los resultados de ENLACE aplicada a inicios del 2011 a 14 millones de niños de

primaria a secundaria, revelan que más de nueve millones de estudiantes tienen un insuficiente nivel en aprendizaje en matemáticas (<http://www.enlace.sep.gob.mx/>).

La organización del artículo es la siguiente: en la sección 2, describimos la Arquitectura del Sistema de Software de nuestra red de aprendizaje. En la sección 3 presentamos la estructura del sistema tutorial afectivo. Los resultados y pruebas de la aplicación en estudiantes son mostrados en la sección 4 y las conclusiones y trabajos futuros son discutidos en sección 5.

2. Arquitectura de la red social y el ITS

Nuestra red social de aprendizaje tiene la funcionalidad básica de toda red social, pero su principal característica es que incluye un STI que ofrece el contenido de un curso en un estilo personalizado para los estudiantes, como se muestra en la Figura 1.

Cada usuario de la red tiene asociado información personal, académica y afectiva en un perfil, el cual es obtenido y almacenado en forma estática y dinámica. El perfil estático contiene la información inicial del usuario (ej. Información personal y académica). El perfil dinámico se actualizará de acuerdo a la interacción del usuario dentro de la red y del STI, tomando en cuenta esta interacción aspectos cognitivos y emocionales. De acuerdo a [6], las emociones están cercanamente relacionadas al aprendizaje del estudiante, lo cual representa el factor clave hacia los resultados del estudiante. Los factores cognitivos son obtenidos de acuerdo al historial que obtenemos de resultados de los exámenes del usuario y el estilo de aprendizaje procesado por la red neuronal. Los factores emocionales son obtenidos por sensores que están monitoreando las emociones del usuario.

3. Sistema tutorial inteligente y afectivo

El sistema tutorial inteligente (STI) para la red social adopta el modelo tradicional conocido como arquitectura de cuatro-componentes donde una interfaz de usuario tiene acceso a los tres principales módulos: dominio, estudiante y módulo tutor. La figura 2 muestra la arquitectura completa del STI.

Modulo Dominio: La adquisición del conocimiento y la representación para el modulo del dominio o experto, es un problema mayor el cual manejamos a través de diferentes conceptos relacionados a la Teoría de Espacios del Conocimiento [15]. Esta teoría provee una base sólida para estructurar y representar el dominio del conocimiento para una enseñanza personalizada e inteligente. Se aplican conceptos de teoría combinatoria que usamos para modelar tutores particulares o personalizados de acuerdo a diferentes estilos cognitivos.

El conocimiento de un curso en el STI es estructurado en forma de un árbol el cual contiene capítulos, los cuales están compuestos por temas. La totalidad de todos los nodos en el árbol representa el conocimiento del modulo experto. El modulo del dominio es implementado por medio de documentos XML (JSON específicamente) cuya estructura es mostrada a continuación:

```
<?xml version="1.0" encoding="UTF-8"?>
<domain>
<name> Domain Name </name>
```

```

<chapters>
  <chapter id="1">Chapter1 </chapter>
  <urlDomain>Domain/Chapter1.xml</urlDomain>
  <urlTest>Domain/Test_Chapter1.xml</urlTest>

<chapter id="2">Chapter2 </chapter>
  <urlDomain>Domain/Chapter2.xml</urlDomain>
  <urlTest>Domain/Test_Chapter2.xml</urlTest>

  <chapter id="3">Chapter3 </chapter>
  <urlDomain>Domain/Chapter3.xml</urlDomain>
  <urlTest>Domain/Test_Chapter3.xml</urlTest>

</chapters>
</domain>

```

Cada capítulo puede tener un archivo XML el cual contiene información específica de cada capítulo y una prueba diagnóstica para el modulo del estudiante.

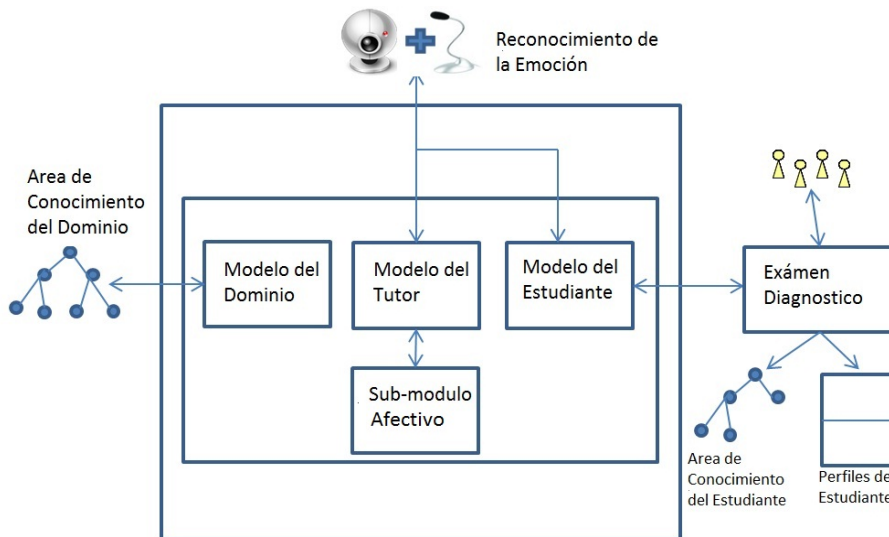


Fig. 2. Arquitectura del STI Afectivo.

Modulo del Estudiante. Este modulo es responsable de evaluar el desempeño del estudiante para determinar habilidades cognitivas y de razonamiento. Provee la información sobre las competencias y capacidades de aprendizaje del estudiante. El STI se da cuenta del conocimiento del estudiante a través de un examen diagnóstico. Los resultados del examen muestran lo que el estudiante conoce y lo que necesita conocer. El modulo de estudiante del STI puede ser visto como un sub-árbol de todo el conocimiento que el experto posee en el dominio, como se muestra en la parte derecha de la figura 2. La representación esta basada en un modelo llamado "Overlay", donde el conocimiento del estudiante es un subconjunto del conocimiento del modulo del dominio. Cuando el estudiante usa el tutor inteligente expande este

subconjunto [16]. Para cada estudiante hay un perfil estático, el cual almacena información particular, académica y un perfil dinámico, el cual almacena información obtenida de la navegación en el tutor y del reconocimiento de emociones.

Cuando un estudiante accede por primera vez al STI, él ha tenido que responder al examen diagnóstico, el cual facilita la construcción del área de conocimiento del estudiante. Dentro de la evaluación diagnóstica cada pregunta tiene un nivel de dificultad y dependiendo del nivel de dificultad, se dan diferentes pesos a las respuestas. Las preguntas difíciles equivalen a 3 puntos, preguntas intermedias valen 2 puntos y las fáciles un punto.

Para determinar la calificación del estudiante aplicamos la siguiente fórmula:

Calificación del Estudiante = Total de Puntos ganados / Suma de los puntos de todas las preguntas

Después de los resultados del examen, un pequeño algoritmo determina el nivel de aprendizaje del estudiante el y el método de enseñanza. A continuación, presentamos el algoritmo para asignar el nivel y método para el aprendizaje del estudiante. El método es elegido de acuerdo con el programa oficial de las Escuelas Públicas de México.

```
Si (Calificación < 4) Entonces
    Nivel = fácil;    Método = Lattice
SiNo
Si (Calificación < 5) Entonces
    Nivel = fácil;    Método = tradicional
SiNo
Si (Calificación < 9) Entonces
    Nivel = Normal;   Método = tradicional
SiNo
    Nivel = Difícil;   Método = tradicional
```

Para la representación del conocimiento del estudiante usamos dos categorías:

- **Temas.** Cada vez que el estudiante toma un tema, lo almacenamos en el historial de temas.
- **Experiencia del Estudiante.** Almacena el historial de las calificaciones por materia.

Ambos están basados en el modelo “Overlay” el cual nos permite conocer el subconjunto de conocimiento que el estudiante posee.

Modulo Tutor. El modulo tutor del STI está basado principalmente en la Teoría de Cognición de ACT-R [3]. Estos tipos de sistemas tutoriales son llamados Tutores de Modelo-Seguimiento o tutores cognitivos. En nuestro caso implementamos las reglas de producción (memoria procedural) y hechos (memoria declarativa) por medio de conjuntos de reglas XML (JSON). Además, desarrollamos un nuevo algoritmo de seguimiento del conocimiento, el cual está basado en lógica difusa, donde para seguir los estados cognitivos del estudiante, se aplica un conjunto de reglas difusas para probar la validez de los hechos. El beneficio de usar lógica difusa es que estas permiten inferencias incluso cuando las condiciones son solo parcialmente

satisfechas. En seguida, presentamos un conjunto de reglas de producción para la operación de división las cuales están escritas en formato JSON:

```

División (
  {"divisor":9,"dividendo":[1,0,8],"cociente":[0,1,2],"residuo":[1,0],"mul":[9,18]},
  {"divisor": 2,"dividendo":[4,2],"cociente":[2,1], "residuo":[0,0], "mul":[4,2]},
  {"divisor": 11,"dividendo":[1,0,0],"cociente":[0,0,9], "residuo":[1], "mul":[99]},
  {"divisor": 10,"dividendo":[5,0,0],"cociente":[0,5,0], "residuo":[0,0], "mul":[50,0]},
  {"divisor": 5,"dividendo":[7,2,5],"cociente":[1,4,5], "residuo":[2,2,0], "mul":[5,20,25]},
  {"divisor": 20,"dividendo":[1,1,2],"cociente":[0,0,5], "residuo":[019], "mul":[100]},
  {"divisor": 2,"dividendo":[4,0,9],"cociente":[2,0,4], "residuo":[0,0,0], "mul":[4,0,8]},
  {"divisor": 20,"dividendo":[5,0,2,0],"cociente":[0,2,5,1], "residuo":[10,2,0], "mul":[40,100,20]},
  {"divisor": 14,"dividendo":[1,3,2],"cociente":[0,0,9], "residuo":[6], "mul":[126]},
);
  
```

Tabla 1. Valores difusos para la variable Dificultad

	Dificultad (%)	Valores Normalizados
Muy Fácil	0% - 10%	0 – 0.1
Fácil	0% - 30%	0 – 0.3
Intermedio	20% - 80%	0.2 – 0.8
Difícil	70% - 100%	0.7 – 1.0
Muy Difícil	90% - 100%	0.9 – 1.0

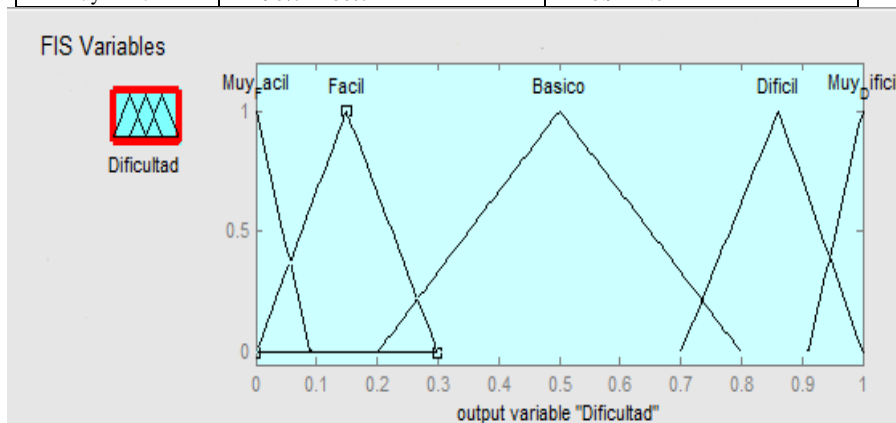


Fig. 3. Conjuntos difusos para la variable “Dificultad”.

Cada tema divide el problema en diferentes categorías de **1** a **n**. El STI lee las reglas y presenta los ejercicios de acuerdo a un nivel de dificultad en el problema. El estudiante no puede moverse al siguiente estado (entrada) a menos que resuelva correctamente todos los ejercicios. Durante esta transición él puede pedir ayuda y en caso de fallar, un mensaje de error es desplegado para ayudar a descubrir cual es la respuesta correcta. Una vez que el estudiante completa el ejercicio, el perfil del estudiante es actualizado con la información en el tipo y dificultad del ejercicio, la cantidad de fallos, ayudas y el tiempo que le tomó resolver el ejercicio. Estas variables (Dificultad, ayuda, errores y tiempo) serán requeridos para determinar el siguiente problema que el estudiante hará. Para esto implementamos un Sistema Difuso Experto que elimina especificaciones arbitrarias de números precisos y crea

decisiones más inteligentes, tomando en cuenta un razonamiento más humano. Los conjuntos difusos son descritos en la tabla 1 y Figura 3 para variable “Dificultad”.

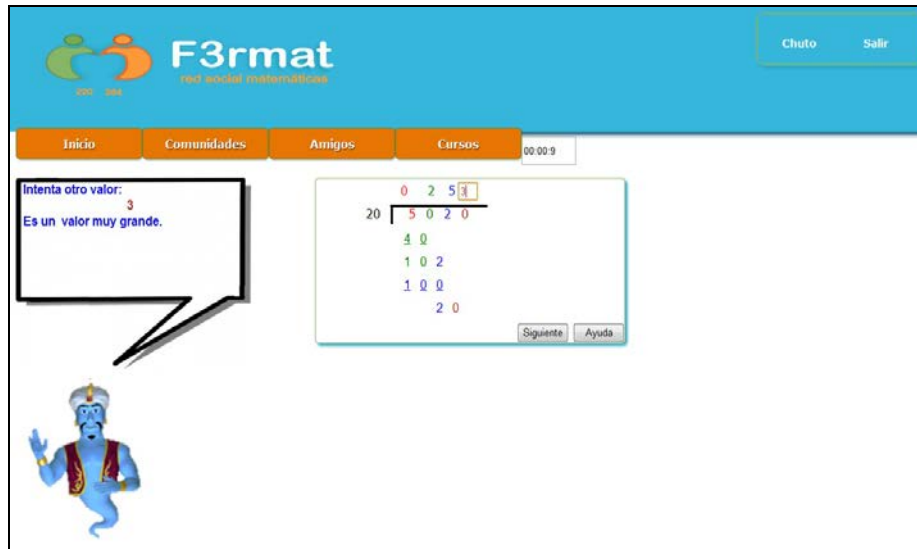


Fig. 4. Respuesta Cognitiva y Afectiva del STI en la Red Social.

A continuación presentamos algunas de las reglas difusas para determinar el grado de dificultad del siguiente ejercicio:

- Si (Error es bajo) & (Asistencia es baja) & (Tiempo es muy rápido) entonces (Dificultad es muy difícil)
- Si (Error es bajo) & (Asistencia es normal) & (Tiempo es lento) entonces (Dificultad es difícil)
- Si (Error es alto) & (Asistencia es alta) & (Tiempo es muy lento) entonces (Dificultad es muy fácil)

Reconocimiento y manejo de Emociones. Las emociones son detectadas por expresiones del rostro y la voz. El método usado para la detección de emociones visuales está basado en la teoría de Ekman [17], la cual reconoce 10 emociones. Nosotros solo buscamos reconocer 7 emociones: ira, disgusto, miedo, felicidad, tristeza, sorpresa y neutral. Para determinar la emoción, primero tomamos la imagen la cual es transformada a una forma más básica. Basado en esta imagen obtenemos los puntos del rasgo que minimizan el conjunto de información de entrada a la red neuronal. Usamos una red neuronal de Kohonen con 20X20 neuronas de entrada y dos de salida representando la emoción. Para cada detección de emociones en la voz, esta es capturada primeramente a través de un micrófono de computadora y luego es normalizada. Entonces aplicamos la técnica para personalizar componentes de análisis (PCA) a la señal representando la voz. Después de usar el método de SFFS [18] obtenemos un conjunto óptimo de características que alimentan la red neuronal. Cada red neuronal usada para reconocer emociones produce una salida. Todas las salidas de

cada red neuronal son integradas utilizando lógica difusa la cual buscamos nos dé un resultado final que es la emoción del usuario que el sistema reconocerá.

Una vez que el estado afectivo del estudiante es determinado, el sub-modulo afectivo tiene que responder congruentemente. Para hacer esto, el tutor necesita un modelo el cual establece parámetros que habilitan un mapeo del estado afectivo y de conocimiento del estudiante para las acciones de enseñanza. Las acciones del tutor inteligente entonces son compuestas por un componente cognitivo y afectivo obtenido de problemas resueltos por el estudiante y sus emociones. El componente afectivo de una acción de enseñanza trata de promover un estado afectivo positivo en el estudiante y el componente cognitivo apunta a transmitir conocimiento. La Figura 4 muestra una interface del ITS dentro de la red social con un agente afectivo representado por el personaje Genio.

4. Resultados y discusiones

La red social junto con el STI fue evaluada por un grupo de niños de tercer grado de Primaria. Hubo 18 niños quienes probaron la herramienta y el sistema tutorial. Antes de la evaluación ofrecimos una pequeña introducción a la herramienta. Evaluamos el tema de la multiplicación y la división. Después de la evaluación aplicamos un examen y un cuestionario al grupo. Con respecto a los resultados, la gráfica en la Figura 5 nos muestra los resultados de 9 niños (escogidos al azar) que usaron el ITS. Podemos ver en los resultados la diferencia de antes y después de usar el ITS para resolver los ejercicios planteados. Las más grandes diferencias son vistas en estudiantes con un bajo nivel inicial para resolver problemas.

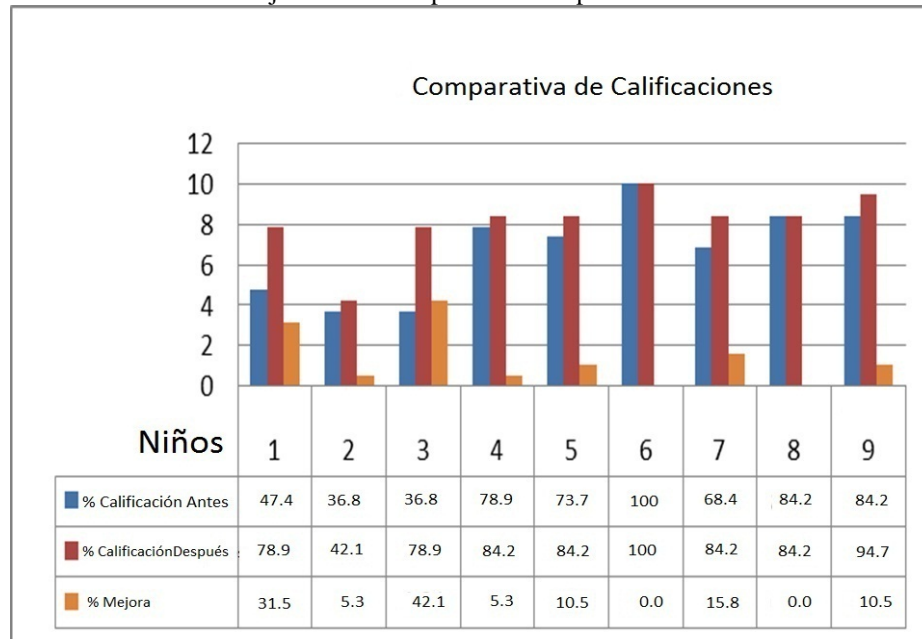


Fig. 5. Resultados de estudiantes usando la Red Social y el ITS.

5. Conclusiones y trabajo futuro

La red social y el ITS fueron implementados usando diferentes herramientas de software y lenguajes de programación. La capa de presentación de la red social fue implementada con CSS3, HTML 5 y JavaScript. Para la capa lógica (tutor inteligente principalmente) usamos Java y JSP. Para la capa de datos usamos XML, JSON y MySQL.

Aun estamos trabajando con el reconocimiento de emociones y estamos agregando más operaciones matemáticas al STI. Sin embargo, los resultados iniciales son motivantes. Estamos implementando nuestros propios reconocedores debido a que necesitamos usarlos en una plataforma Web donde la red social puede ser accedida desde diferentes equipos como laptops y teléfonos inteligentes (smart-phones).

Agradecimiento. El trabajo descrito en este documento fue financiado por DGEST.

Referencias

1. Carbonell, J. R.: AI in CAI: An artificial intelligence approach to computer-aided-instruction. *IEEE Transactions on Man-Machine System*. MMS, 11(4), 190-202 (1970)
2. Clancey, W. J.: Transfer of rule-based expertise through a tutorial dialogue. *Computer Science Stanford, CA, Stanford University*. PhD. (1979)
3. Anderson, . R., Boyle, C. F., Corbett, A. T., & Lewis, M. W.: Cognitive modeling and intelligent tutoring. *Artificial Intelligence*, 42, 17-49. Doi: 10.1016/0004-3702(90)90093-F (1990)
4. Aleven, V., Koedinger, K: An effective metacognitive strategy: learning by doing and explaining with a computer-based cognitive tutor. *Cognitive Science* 26(2), 147-179 (2002)
5. Woolf, Beverly Park: Building intelligent interactive tutors. Morgan Kaufmann. ISBN978-0-12-373594-2 (2009)
6. Picard, W.R.: *Affective Computing*. The MIT Press. Cambridge, Massachusetts (1997)
7. Arroyo, I., Woolf, B., Cooper, D., Bursleson, W., Muldner, K., Christopherson, R.: Emotions sensors go to school. In: Diminitrova, V., Mizoguchi, R., Du Boulay, B., Graesser, A. (eds.) *Proceedings of the 14th international conference on artificial intelligence in education*, pp. 17-24. IOS press, Amsterdam (2009)
8. Conati, C., McLaren, H.: Evaluating a probabilistic model of student affect. In: Lester, J.C., Vicai, R. M., Paraguacu., F. (eds.) *ITS 2004. LNCS*, vol. 3220, pp. 55-66. Springer, Heidelberg (2004)
9. D’Mello, S.K., Picard, R.W., Graesser, A.C.: Towards an affective-sensitive AutoTutor. *Special issue on Inteligent Educational Systems IEEE Intelligent Systems* 22(4), 53-61 (2007)
10. Essa, I. A. and Pentland, A.: Facial expression recognition using a dynamic model and motion energy. In: *IEEE international conference on computer vision*, pp. 360-367, Cambridge, MA, Computer society (1995)
11. Yacoob, Y. and Davis, L.S.: Recognizing human facial expressions from log image sequences using optical flow. *IEEE Transactions on pattern analysis and machine intelligence*, 18(6):636-642 (1996)
12. Tosa, N. and Nakatsu, R.: Life-like communication agent-emotion sensing character ‘MIC’ and feeling session character ‘MUSE’. In: *Proceedings of the international conference on multimedia computing and systems*, pp. 12-19, IEEE (1996)

13. Graesser, A., Lu, S.L., Jackson, G., Mitchell, H., Ventura, M., Olney, A., Louwerse, M.: AutoTutor: A tutor with dialogue in natural language. *Behavioral research methods, instruments, and computers* 36, 180-193 (2004)
14. Du Boulay, B.: Towards a motivationally intelligence pedagogy: how should an intelligent tutor respond to the unmotivated or the demotivated? *New perspectives on affect and learning technologies, Explorations in the learning sciences, instructional systems and performance technologies* 3, pp. 41-52, Springer science+business media (2011)
15. Doignon, J. -P. and Falmagne, J. C. *Knowledge Spaces*. Springer-Verlag (1999)
16. Günel, K.: *Intelligent Tutoring Systems: Conceptual Map Modeling*. Lambert Academic Publishing (2010)
17. Ekman P, Oster,H.: Facial expressions of emotion. *Annual Review of Psychology* 30:527-554 (1979)
18. Pudil P, Novovičová J, Kittler J.: Floating search methods in feature selection. *Pattern Recognition Letters* 15 (11) (1994)
19. Hernández, Y., Sucar, L.E., Arroyo G.: Building an affective model for intelligent tutoring systems with base on teachers' expertise, pp 754-764, Springer-Verlag (2008)
20. Clemen, R.T.: *Making hard decisions*, Duxbury Press, Belmont CA (2000)

Sistema de red inteligente Mindmapping (SRIM)

Rosa María Rodríguez Aguilar, Sabino Miranda-Jiménez y Doricela Gutiérrez Cruz

Universidad Autónoma del Estado de México, Unidad Académica Profesional Nezahualcóyotl,
Estado de México, México

{rmrodriguez,smirandaj}@uamex.mx, gutierrezcruzdo@yahoo.com.mx

Resumen Durante décadas los investigadores se han enfocado en mejorar y potenciar la capacidad de aprendizaje del alumno, bajo la tendencia de los nuevos paradigmas dentro de la dinámica del conocimiento. Por lo cual, se propone el diseño de un sistema tutorial adaptable de acuerdo a los conocimientos previos y a la capacidad de evolución de aprendizaje del alumno. El sistema se modela con base a las redes neuronales artificiales las cuales proporcionan la arquitectura de evaluación general con ayuda de los mapas mentales (Mindmapping) que codifican el conocimiento del experto.

Palabras clave: Mapas mentales, redes neuronales, sistemas tutoriales inteligentes.

1. Introducción

Los cambios tecnológicos y sociales de los últimos años han producido un fuerte impacto en los ambientes educativos. Las competencias necesarias para el desenvolvimiento de los individuos en la sociedad, junto con las posibilidades que las tecnologías de la información y de la comunicación (Tics) presentan para los procesos de enseñanza-aprendizaje, interpelan las prácticas docentes, requiriendo su revisión y reflexión sobre su integración en los proyectos pedagógicos [1].

Dentro de las tecnologías en los procesos de enseñanza-aprendizaje los Sistemas Tutoriales Inteligentes van adquiriendo mayor importancia, ya que pretenden tener la habilidad de imitar a los tutores humanos para determinar en cada caso qué enseñar, cuándo enseñar y cómo enseñar de un modo autónomo.

A pesar, de que el uso de la computadora en la educación viene desde los años sesentas, es a partir del nacimiento de la microcomputadora cuando surge un verdadero auge en el uso de la computadora en la enseñanza de la matemática.

El software educativo aparece con la difusión de las computadoras en la enseñanza, bajo tres sistemas de trabajo: utilizadas como tutores (enseñanza asistida por computadoras (CAI), integrando la inteligencia artificial (ICAI) y finalmente como sistemas tutoriales inteligentes (STI). Los STI difieren de los sistemas anteriores en que en su implementación agregan un componente más, el cual contiene las estrategias pedagógicas (sugerencias del sistema hacia el alumno, llamadas de atención) necesarias para que el estudiante obtenga un conocimiento significativo [9].

En los años setenta Tony Buzan [2] se planteó si era posible o no incrementar el nivel de la inteligencia en las personas. Por lo que desarrolló varias técnicas, una de ellas fueron los mapas mentales (*Mindmapping*, del inglés). Con el uso de los mapas mentales, las personas incrementaron sus coeficientes de inteligencia, mejoraron la memoria y las aptitudes de pensamiento reforzado.

En este artículo nos enfocamos en el diseño de un esquema que mida la cantidad de conocimiento aprendido por el alumno en relación al conocimiento de un experto humano en un dominio específico en el marco de un Sistema Tutorial Inteligente. Se hace uso de mapas mentales para representar el conocimiento del experto y las redes neuronales para la evaluación del conocimiento.

2. Sistemas tutoriales inteligentes (STI)

Aunque no existe una definición formal de lo que es un Sistema Tutorial Inteligente (STI), la mayoría de los investigadores coincide en que son sistemas instruccionales, adaptativos a los estudiantes, y que procuran emular los beneficios del proceso de enseñanza-aprendizaje tradicional. Los STIs tienen como objetivo imitar a los tutores humanos en su habilidad para determinar en cada caso qué enseñar, cuándo enseñar y cómo enseñar de manera autónoma.

Un STI usa tres tipos de conocimiento: conocimiento del dominio (tópicos de una materia en particular), conocimiento de estrategias y métodos de enseñanza (el método de guía particular o *coaching*), y conocimiento sobre el estudiante (preferencias y estilos de aprendizaje personales). Estos tres tipos de conocimiento proveen al sistema la capacidad de asistir al aprendiz.

Actualmente los STIs son espacios en donde se crean las condiciones adecuadas para que el individuo se apropie de nuevos conocimientos, nuevos elementos que le sugieren procesos de análisis y reflexión [1, 2].

Generalmente, un STI debe identificar las fortalezas y debilidades de un estudiante particular a fin de establecer un plan instruccional que será consistente con los resultados obtenidos. Debe encontrar la información relevante sobre el proceso de aprendizaje de ese estudiante (como estilo de aprendizaje) y aplicar el mejor medio de instrucción según sus necesidades individuales.

Para apoyar el proceso de aprendizaje, algunos autores han utilizado mapas de conceptos [8] para la representación del conocimiento del experto dentro de los STI. Por lo que asumen que el estudiante asimilará el conocimiento estructural del experto a través de la navegación de los tópicos en la estructura conceptual de un dominio específico [9].

Sin embargo, no proponen ninguna forma para evaluar el aprendizaje del estudiante en tales estructuras conceptuales. Nosotros proponemos un esquema que hace uso de los mapas mentales (similar a los mapas de conceptos) y las redes neuronales para llevar a cabo la evaluación del nivel de aprendizaje del estudiante con relación a los temas presentados en el sistema.

2.1. Mapas mentales

Un mapa mental [2] es un diagrama usado para representar las palabras, ideas, tareas, u otros conceptos, los cuales están relacionados con la idea central o palabra clave de algún tema específico. Estos diagramas son similares a una red semántica sin delimitaciones formales en las clases de enlaces usados para unir los conceptos.

Los elementos en el mapa se arreglan intuitivamente según la importancia de los conceptos y se organizan en agrupaciones de diferentes niveles. Esta técnica intenta modelar la memoria del ser humano, ya que es una forma dinámica de organizar las ideas que van surgiendo en el cerebro.

Se parte de la consideración de que los mapas mentales se visualizan como grafos representando la lluvia de ideas que arroja alguna temática por analizarse. Los grafos indican el avance de un estado de conocimiento a otro mediante aristas o conexiones de situaciones cognitivas caóticas hacia un proceso ordenado. La organización es apoyada por una perspectiva pedagógica en la que se trabaja algún tema de nivel mental caótico al terreno conceptual ordenado y documentado. De tal manera que se obtenga un aprendizaje significativo de acuerdo a los resultados guiados por los mapas mentales los cuales aportan un medio sistemático para registrar y estimular el flujo natural del proceso de pensamiento al crear un circuito cerrado de aprovechamiento cognitivo entre el cerebro y los mapas mentales.

2.2. Red neuronal

Una red neuronal artificial intenta modelar el comportamiento de una red neuronal biológica; está formada por un conjunto de unidades computacionales simples que están altamente interconectadas. Las conexiones de una unidad a otra sirven para propagar la activación de la neuronas [7].

En la Figura 1, se describen los elementos básicos que intervienen en una red neuronal.

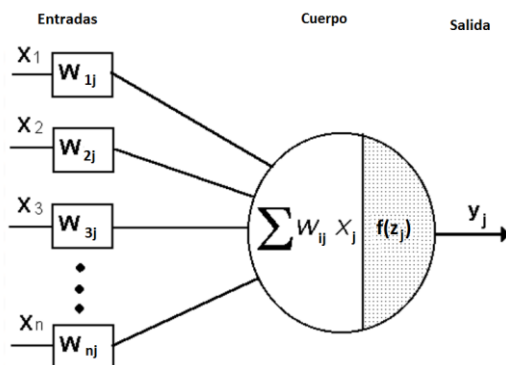


Fig. 1 Elementos característicos de una red neuronal.

La red neuronal está formada por tres componentes principales: los datos de entrada codificados (X_n) y pesos (W_{ij}), la función de activación o transferencia y una salida

que representa el grado de similitud entre la información de entrada y la información de entrenamiento. La función $f(z_j)$ es llamada la función de activación, y es la función que utiliza la suma de estímulos para determinar la actividad de salida de la neurona.

Una de las principales ventajas las redes neuronales es la simplicidad de sus algoritmos para aprender a partir de ejemplos o datos de entrada [6,8].

En este artículo, se presenta el diseño de un sistema tutorial inteligente en el área de la matemática, en especial en álgebra, que modele el conocimiento del experto humano por medio del Sistema de Red Inteligente Mindmapping (SRIM), el cual permite evaluar los conocimientos de cada estudiante con relación al experto humano.

3. Arquitectura del sistema tutorial

La arquitectura general del Sistema Tutorial Inteligente se muestra en la Figura 2 [10].

El sistema se constituye de los módulos *Experto*, *Estudiante*, *Tutorial* e *Interfaz*. *Módulo Experto*: contiene el conocimiento que el sistema le presenta al alumno. *Modelo del Estudiante*: captura el entendimiento del aprendiz sobre el dominio de estudio. *Módulo Tutorial*: contiene las estrategias tutoriales e instruccionales. Estas estrategias deben ajustarse a las necesidades del estudiante sin la intervención del tutor humano. *Módulo Interfaz*: es el entorno de simulación del espacio de enseñanza-aprendizaje.

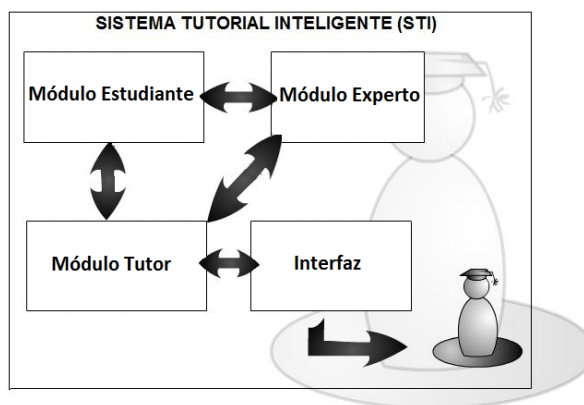


Fig. 2. Arquitectura del Sistema Tutorial.

En la arquitectura presentada en [10] retomamos el diseño del Módulo Tutorial el cual implementa el mecanismo de evaluación contrastando los conocimientos del experto y los conocimientos del estudiante por medio de medidas probabilísticas.

El modelo de evaluación que proponemos, Red Inteligente Mindmapping (SRIM), se basa en una red neuronal la cual contrasta los mapas mentales del experto contra los mapas mentales del alumno generados automáticamente por el sistema.

El modelo de la red neuronal a utilizarse es el Perceptrón Simple, dadas las características probabilísticas del mismo. Una de las características importantes es aprender a reconocer patrones. El Perceptrón está constituido por conjunto de sensores de entrada que reciben los patrones de entrada a reconocer o clasificar y una neurona de salida que se ocupa de clasificar a los patrones de entrada.

En nuestro modelo, la red neuronal aprende los patrones de los mapas mentales generados por el experto en el área de estudio (la matemática). Posteriormente, la red neuronal toma como estradas los mapas mentales del alumno generados por el sistema y los compara contra los mapas del experto proporcionando una medida de similitud entre los conocimientos el experto y los del alumno a nivel conceptual.

Por un lado, los mapas mentales del experto representan las soluciones de los problemas, en el ámbito de estudio (álgebra básica). Los mapas mentales son realizados por el experto humanos especializados en matemáticas y apoyado por pedagogos que determinan la mejor estrategia de solución. El experto procede a resolver un problema sobre un tópico y genera su solución en una red conceptual, la cual contiene la representación del conocimiento que se debe adquirir para el tópico en cuestión.

Actualmente, se cuenta con 10 reactivos prototípicos en el área de álgebra. Se cuenta con la representación en forma de mapa mental de tales reactivos.

Por otro lado, los mapas mentales del alumno se generarán a través de la interacción del alumno con el sistema tutorial. Para su generación se consideran los errores y aciertos del alumno durante la resolución de reactivos; y los tópicos por aprender en cada módulo, los cuales aumentarán su dificultad conforme avance el alumno en conocimiento dentro sistema.

Con los conjuntos de mapas mentales del alumno y experto se realizará la evaluación del conocimiento aprendido por el alumno por medio de la similitud de ambos esquemas por medio de la red neuronal. Esta medida de similitud también representa el grado de dificultad del tópico, ya que los temas están preparados pedagógicamente y un valor bajo indicaría que el tópico es complejo.

Los datos proporcionados por la red neuronal será almacenados en la base de conocimiento del sistema, los cuales servirán para que el sistema se ajuste y realimente al alumno con sugerencias de estudio y repaso de temas específicos.

4. Conclusiones

Los mapas mentales como técnica para representar el conocimiento han dado buenos resultados a las personas al incrementar su capacidad de aprendizaje. Por lo cual, se propuso un método para evaluar el conocimiento adquirido por el alumno considerando la representación del conocimiento en forma de mapas mentales.

El método propuesto evalúa el conocimiento adquirido del alumno contra el conocimiento del experto modelado como mapa mental por medio de una red neuronal. Actualmente, se ha desarrollado un conjunto de temas, sobre álgebra, representados en forma de mapas mentales (10 tópicos).

Referencias

1. Ausubel, D., Novak, J., Hanesian, H.: *Psicología Educativa: Un punto de vista cognoscitivo*, Editorial Trillas, México (1983)
2. Buzan, T.: *El libro de los mapas mentales. Cómo utilizar al máximo las capacidades de la mente*. Urano, Barcelona Tit.Orig: *TheMindMap Book*, BBC Worldwide Publishing (1996)
3. Hertz, J., Krogh, A., Palmer, R.G: *Introduction to the theory of neural computation*. Addison-Wesley, Redwood, California (1991)
4. Rumelhart, D. E., Ortony, A.: *The representation of know ledge in memory*. En: Anderson, R.C., Spiro, R.J., Montague, W. E.W (eds.) *Schooling and the acquisition of knowledge*. Lawrence Erlbaum , Nueva jersey (1977)
5. Rumelhart, D. E., Mclelland, J.L.: *Grupo PDP. Introducción al procesamiento distribuido en paralelo*. Alianza, Madrid (1992)
6. Russell, P., Norving, P.: *Inteligencia Artificial un enfoque moderno*. Prentice – Hall, EUA (1996)
7. Jonassen, D., Reeves, T., Hong, N., Harvey, D. Y Peters, K.: *Concept Mapping as Cognitive Learning and Assessment Tools*. *Journal of Interactive Learning Research*. (8) 289-308 (1997)
8. Huapaya, C.R., Arona, G.M., Lizarralde, F.A.: *Sistemas Tutoriales Inteligentes Aplicados a Dominios de la Ingeniería. JEITICS 2005 - Primeras Jornadas de Educación en Informática y TICS en Argentina* (2005)
9. Self, J.: *The defining characteristics of intelligent tutoring sytems research: ITSs care, precisely*. In *International Journal of Artificial Intelligence in Education*. V. 10, pp. 350-364 (1999)
10. Rodríguez, R.M., Laureano, A.L., Sánchez de Antuñano, J.M.A.: *Diseño de un Ambiente Virtual de Aprendizaje: un enfoque reactivo a las distintas inteligencias*. XXIII Congreso Nacional y XI Congreso Internacional de Informática y Computación de la ANIEI, Puerto Vallarta 11-15. ISBN 978-607-707-097-9, pp. 494-505 (2010)

Editorial Board of the Volume
Comité editorial de volumen

Grigori Sidorov
Raúl Monroy
Alexander Gelbukh
Miguel González-Mendoza
Ildar Batyrshin
Oscar Herrera-Alcántara
Sofía Galicia-Haro
Félix Castro-Espinosa

Author Index Índice de autores

Trejo, Luis A.	199, 297	Ibarra-Manzano, Mario-Alberto	271
Aguilar-Galicia, Honorato	15	Olmos, Iván	37
Alemán, Yuridiana	57	Jaimez-González, Carlos R.	177
Almanza-Ojeda, Dora-Luz	271	De la Torre-Gea, Guillermo	249
Alonso, Roberto	199	Laureano Cruces, Ana Lilia	151
Alvarado V., Alejandra	321	Ledeneva, Yulia	15, 113
Aranda Benítez, Boris A.	365	López Arizmendi, J.M. de Jesús	377
Aravena A., Rodrigo	69	López-López, Aurelio	27
Barajas, Noemí	333	Loya, Nahun	57
Barrón-Estrada, María L.	377	Luis Carballido, José	209
Benito Camiña, J.	285	Luna-Rodríguez, Francisco	271
Caballero Morales, Santiago Omar	81	Madrid Hurtado, Luis Arturo	187
Carpio, Martín	221	Meneses Mendoza, Sonia Regina	125
Carrillo, Maya	27, 37	Miranda-Jiménez, Sabino	387
Castillo, Esteban	37	Monroy, Raúl	199, 285, 297
Castro-Manzano, José Martín	165	Morales, Rocío	113
Contreras A., Ricardo	69, 321	Morales-Palafox, Edgar	3
Cruz L., Roberto O.	239	Olivares Camacho, José Luis	377
Cuautle Rivera, Norma L.	27	Orozco-Rosas, Ulises	353
Sanchez De Antuñano, Jorge	151	Ortiz-Gaucin, Angélica	343
Delfín-Santiesteban, Oscar	249	Osorio Galindo, Mauricio	259
Domínguez Pérez, Oscar	151	Pérez Flores, Esteban	271
Estrada Benavides, Roberto J.	365	Pérez-y-Pérez, Rafael	3
García-Vázquez, Mireya	311, 353	Picos-Espinoza, Kenia	311
Gelbukh, Alexander	45	Pineda Torres, Ivo H.	103
Gómez Adorno, Helena	57	Pinninghoff J., M. Angélica	69, 321
González Trapero, Karen Yadira	377	Pinto, David	37, 57
González-Marrón, David	137	Priego Sánchez, Belém	103
González-Mendoza, Miguel	93, 137	Puga-Soberanes, Héctor José	221, 239
Gutiérrez Cruz, Doricela	387	Ramírez-Acosta, Alejandro A.	311, 353
Hernández Gómez, Jaime	103	Ramos-Martínez, Marina I.	221
Hernández Santos, David Isaac	271	Rodríguez Aguilar, Rosa María	387
Hernández-Gracidas, Carlos	285	Rodríguez, Daniel	209
Hernández-Gress, Neil	93, 137		
Huitzil Tello, Adriana	259		

Rodríguez-Elías, Oscar Mario	125, 187	Trujillo Romero, Felipe	81
Rose Gómez, César Enrique	125	Valencia Palomo, Guillermo	187
Ruiz Domínguez, Germán Alonso	187	Vázquez, Jorge	297
Ruiz-Castilla, Sergio	113	Velázquez Mendoza, María de Jesús	125
Santiago M., Raúl	239	Vilariño, Darnes	37
Santiago Santiago, Karina	151	Villagrán-Olguín, Alfredo	93
Santiago, Raul	343	Villavicencio Gómez, Laura	365
Santiago-Montero, Raul	333	Suárez Y Gómez, J.L. Fernando	333
Sarmiento Bustos, Estela	151	Zamarrón, Antonio	343
Sidorov, Grigori	15	Zatarain-Cabada, Ramón	377
Somodevilla García, María J.	103	Zepeda Cortés, Claudia	209, 259
Soria-Alcaraz, Martín	221	Zhila, Alisa	45
Tovar Rodríguez, Julio César	231		

Impreso en los Talleres Gráficos
de la Dirección de Publicaciones
del Instituto Politécnico Nacional
Tresguerras 27, Centro Histórico, México, D.F.
Junio de 2012
Printing 500 / Edición 500 ejemplares