Avances en la Ingeniería del Lenguaje y del Conocimiento

Research in Computing Science

Series Editorial Board

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. **Volumen 124,** noviembre 2016. Tiraje: 500 ejemplares. *Certificado de Reserva de Derechos al Uso Exclusivo del Título* No.: 04-2005-121611550100-102, 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. 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 responsable: *Grigori Sidorov, RFC SIGR651028L69*

Research in Computing Science is published by the Center for Computing Research of IPN. **Volume 124**, November 2016. Printing 500. The authors are responsible for the contents of their articles. Printed in Mexico City, in the IPN Graphic Workshop – Publication Office.

Volume 124

Avances en la Ingeniería del Lenguaje y del Conocimiento

David Pinto
Darnes Vilariño
Beatriz Beltrán (eds.)

Instituto Politécnico Nacional, Centro de Investigación en Computación
México 2016

ISSN: 1870-4069

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.rcs.cic.ipn.mx
http://www.ipn.mx
http://www.cic.ipn.mx

Indexed in LATINDEX, DBLP and Periodica

Printing: 500

Printed in Mexico

ISSN 1870-4069

Editorial

Esta edición especial de la Revista Research in Computing Science contiene una serie de contribuciones originales que han sido seleccionadas a partir de un proceso de evaluación ciega doble (double blind), lo cual significa que los nombres de los autores de los artículos y los nombres de los revisores son ambos desconocidos. Este procedimiento es ejecutado en aras de proveer una evaluación anónima, que derive en artículos de mayor calidad para este volumen; particularmente, en esta ocasión la tasa de rechazo fue del 34%, cuidando que en todos los casos, al menos dos especialistas del comité revisor hicieran una evaluación de la pertinencia, originalidad y calidad de cada artículo sometido.

Las contribuciones presentes en este volumen son el resultado de una selección de los mejores artículos que fueron previamente presentados en el simposio en Ingeniería del Lenguaje y del Conocimiento (LKE‘2016), en particular en la cuarta edición de esta serie de eventos. Esta conferencia ha sido organizada en el seno de la Facultad de Ciencias de la Computación de la Benemérita Universidad Autónoma de Puebla (BUAP) por cuatro años consecutivos, y nace como una iniciativa del laboratorio de Ingeniería del Lenguaje y del Conocimiento con la finalidad de ofrecer un espacio académico y de investigación, en el cual sea posible reportar trabajos relacionados con el área. Este evento promueve la cooperación entre diferentes grupos de investigación, pues permite el intercambio de resultados científicos, prácticos y la generación de nuevo conocimiento.

Esperamos que este volumen sea de utilidad para el lector y los autores de los artículos seleccionados encuentren en esta edición especial un espacio de intercambio científico productivo que enriquezca la colaboración entre estudiantes y académicos en el ámbito de la ingeniería del lenguaje y del conocimiento.

Deseamos agradecer a la Red Temática en Tecnologías del Lenguaje y a la Sociedad Mexicana de Inteligencia Artificial por los apoyos brindados.

El proceso de revisión y selección de artículos se llevó a cabo usando el sistema libremente disponible llamado EasyChair, http://www.easychair.org.

David Pinto
Darnes Vilariño
Beatriz Beltrán

Facultad de Ciencias de la Computación,
LKE-FCC-BUAP, México

Editores invitados

Noviembre 2016

ISSN 1870-4069

Table of Contents

Page

ISSN 1870-4069

Representación semántica de eventos sobre seguridad: un enfoque basado en lingüística

José A. Reyes-Ortiz

Universidad Autónoma Metropolitana,
Departamento de Sistemas, México

jaro@correo.azc.uam.mx

Resumen. Este artículo reporta la elaboración de una ontología mediante un proceso basado en el principio de modularización. El proceso considera la modularización, la cual consiste en dividir un sistema de ontologías en ontologías individuales con la finalidad de facilitar la reutilización, mantenimiento e integración de ontologías. El principal objetivo de este artículo es diseñar un sistema de ontologías para la representación de eventos sobre seguridad apoyado en consideraciones lingüísticas, el cual se enfoca en la creación de módulos desde el inicio del proceso de diseño de ontologías. El sistema de ontologías es creado de manera modular y evaluado utilizando una fase de axiomatización.

Palabras clave: Eventos sobre seguridad, creación modular de ontologías, representación de conocimiento, aspectos lingüísticos.

Semantic Representation of Security Events: A Linguistic-based Approach

Abstract. This paper reports the development process of an ontology based on the principle of modularization. The purpose of considering a modularization is to divide the whole system of ontologies into individual ontologies in order to facilitate reuse, maintenance and integration of them. Therefore, the main aim of this paper is to design a set of ontologies for representing security-related events, which is supported on linguistic issues and a modular design process is conducted from the beginning of such process. A set of individual ontologies is created in a modular way, they are integrated to form the ontological system, and then, it is evaluated with an instantiation phase.

Keywords. Security events, modular design ontologies, knowledge representation, linguistic aspects.

1. Introducción

La cantidad de textos periodísticos en la Web está creciendo todos los días. Este tipo de textos contienen eventos en la mayoría de sus párrafos, los cuales describen sucesos y un conjunto de dimensiones, a saber: lugar del suceso, tiempo en que ocurrió el suceso, causas y objetos o agentes involucrados. Con la proliferación de este tipo de textos, son necesarias, cada día con mayor urgencia, herramientas de extracción automática, representación y búsqueda de información centrada en eventos. La representación de eventos es un factor clave en este conjunto de herramientas ya que provee los mecanismos para ayudar a la extracción de información sobre eventos en textos periodísticos no estructurados y hacer posible su búsqueda.

Un modelo ontológico se debe considerar para lograr una representación de eventos basada en su significado y con una estructura semántica capaz de lograr inferencias y localización automática de información. Las ontologías son capaces de proveer los mecanismos necesarios para afrontar la problemática descrita arriba, ellas son un modelo computacional que se utiliza para la representación de información de un dominio específico, sus propiedades y como se relacionan los conceptos [1].

Este artículo se centra en resolver la carencia de una estructura semántica procesable por computadoras con información de eventos reales. Esta representación considera el principio de modularización, reportado en [2], en el proceso inicial de construcción de un sistema de ontologías para la representación de eventos sobre seguridad y proporcionar un repositorio estructurado semánticamente de eventos. Este resultado puede ser útil para el análisis automatizado de información sobre eventos y lograr vincular datos de eventos dispersos en la Web.

El resto del artículo está organizado de la siguiente manera. En la Sección 2 se presentan los trabajos relacionados con la representación de información utilizando ontologías. La Sección 3 presenta las teorías sobre la cognición de eventos en general y de manera específica se aterriza en eventos sobre seguridad. La creación del sistema de ontologías para la representación de eventos sobre seguridad basada en aspectos lingüísticos, se muestra en la Sección 4. La axiomatización del sistema de ontologías se expone en la Sección 5, con la finalidad de validar y verificar dicho sistema. Finalmente, las conclusiones y el trabajo a futuro son presentados en la Sección 6.

2. Trabajos relacionados

La tarea de representación de conocimiento ha presentado avances significativos en los últimos años, esto gracias a la incorporación de ontologías como mecanismo de representación. Por ello, en esta sección se presentan los trabajos relacionados con la representación de eventos utilizando ontologías.

En esta área de investigación, se han presentado diversos trabajos como en [3], donde se expone un método para la extracción de eventos violentos a partir de noticias en línea con el propósito de instanciar una ontología, en el cual se utilizan patrones textuales y agrupamiento de textos. También, es el caso del trabajo presentado en [4] que desarrolla

una ontología del dominio de seguridad de la información para describir y representar eventos ocasionados por ataques cibernéticos o software maliciosos.

En [5] se reporta el proceso de construcción de una ontología para resolución de identidades en el dominio de la salud pública mediante la representación de eventos de nacimientos, decesos, certificados de registros y reportes mixtos de salud. La idea de los autores es solucionar los problemas de heterogeneidad estructural y semántica al vincular fuentes de datos dispersas, utilizando una ontología que funcione como un repositorio semántico, el cual proporcione una visión de cómo la identidad de un individuo evoluciona con el tiempo.

Por su parte, en [6] se describe un sistema que reconoce eventos a partir de noticias. El sistema presentado clasifica noticias y genera instancias de clases definidas en una ontología creada manualmente. Los autores crean una topología de eventos, con relaciones semánticas entre eventos y propiedades, para posteriormente, desempeñar un proceso de extracción de conocimiento con la finalidad de poblar la ontología.

El trabajo presentado en [7] tiene por objetivo el descubrimiento y representación semiautomático de eventos financieros utilizando patrones léxicos-semánticos a partir de textos. Sin embargo, los autores no crean una ontología con los datos de eventos descubiertos, por su parte, ellos crean una base de conocimiento con la información de eventos extraída de textos sobre finanzas.

En el dominio de la medicina, específicamente, en el dominio de fármacos, diversos trabajos han propuesto la representación, extracción y recuperación de eventos adversos provocados por los fármacos utilizando ontologías para apoyar el proceso de toma de decisiones de los médicos [8, 11].

Este trabajo centra su aportación en la incorporación de la modularización desde el inicio del proceso de construcción de un sistema de ontologías para la representación de eventos sobre seguridad en español. La idea es que este sistema de ontologías apoye la tarea de extracción automática de eventos a partir de noticias en español.

3. Cognición de eventos sobre seguridad

La cognición se refiere a la forma en que la mente (cerebro) conoce y percibe el mundo. Por su parte, los procesos cognitivos hacen referencia a la adquisición de conocimiento a partir de la percepción otorgando un significado y una organización a la información percibida.

El proceso cognitivo ha sido estudiado en diferentes campos incluyendo la neurología, psicología, sociología y filosofía. En la actualidad, estos estudios se han trasladado a las ciencias computacionales y ha despertado un interés particular en la inteligencia artificial, la representación de conocimiento y el aprendizaje automático.

Los eventos intervienen en el proceso cognitivo desde que estos ayudan a organizar la información percibida y otorgan un orden espacio-temporal, además sitúan a los protagonistas que aparecen en dicha información. De esta manera, los eventos han sido concebidos desde el proceso cognitivo, como un suceso que involucra un cambio de estado, donde se involucran aspectos locativos, temporales y causales [12].

Este artículo se centra en la cognición de eventos sobre seguridad, donde los eventos son el núcleo sobre el cual actúan las dimensiones (espacialidad, temporalidad, protagonistas y causalidad), esto se debe a su intervención e importancia en la forma que los humanos perciben y estructuran la información durante el proceso de comprensión del conocimiento (proceso cognitivo).

De esta cognición se obtiene una definición formal del concepto *Evento* y sus elementos, la cual se muestra en la ecuación 1:

$$Evento = (E, S, T, C, P), \tag{1}$$

compuesta por una etiqueta *E* del evento; *S*, *T*, *C* y *P* corresponde a las dimensiones de espacialidad, temporalidad, causalidad y protagonistas. La información de las dimensiones constituye los complementos del evento, los cuales agregan significado y organización a los eventos.

Los eventos son los puntos focales de las situaciones del mundo (textos, imágenes, audio, hechos, entre otras), por lo tanto, al intentar comprenden estas situaciones, el receptor construye una representación de los protagonistas, eventos, estados, acciones, relaciones, espacio, tiempo, causas y efectos.

Los textos en lenguaje natural son una forma de comunicar las situaciones, por lo tanto, se comprensión involucra poner una atención focal en los eventos descritos con la finalidad de crear una representación de lo que se expresa en términos espacio, tiempo, causas y protagonistas. En estos textos, la estructura lingüística actúa como un medio para expresar cómo está construida la situación o el mundo.

Los eventos sobre seguridad, como acaecimientos, están presentes en la mayoría de los textos periodísticos. En estos textos, los eventos están representados por una frase verbal (*L. Tesnière* [13], *M. Halliday* [14]). Tanto la teoría de *L. Tesnière* como la de *M. Halliday* afirman que el verbo es el núcleo sobre el cual giran todos los elementos de la oración, los cuales se dividen en actantes (agente, objeto, y beneficiario) y los circunstantes (instrumento, fuerza, tiempo y locativo).

Los actantes, en los eventos sobre seguridad, responden a preguntas como: ¿quién realizó el hallazgo?, ¿con qué realizó el ataque?, ¿quién fue arrestado?, por mencionar algunas.

La presencia de estos actantes en los eventos sobre seguridad depende en gran medida del tipo de evento y el verbo que lo caracteriza, y por consecuencia, el número de valencias necesarias. Estas valencias cambian de acuerdo al contexto y al significado del verbo. Para percibir esta información, se presenta la teoría de valencias de *Tesnière* ([13]), haciendo un énfasis en los valores actanciales de los verbos.

La teoría de *Tesnière* afirma que la valencia de los verbos es el número de actantes que puede recibir para producir un significado coherente. Esta teoría se aplica a diversas lenguas neolatinas como el español.

El análisis de valencias de los verbos permite comprender oraciones ambiguas. Por lo tanto, todo verbo tiene su propio valor actancial, los cuales se clasifican, según [13], en los siguientes grupos.

a. **Verbos avalantes**. Aquellos verbos que no tienen actantes y su significado no se ve afectado por esta circunstancia, ejemplos de estos verbos son: *llover*,

nevar, *tronar* (verbos de tiempo atmosférico), *ser* (en expresiones como ser tarde, ser necesario, ser lícito), *haber* (con valor impersonal).

b. **Verbos monovalentes.** Verbos que requieren un actante, comúnmente el sujeto sintáctico que puede ser alguien o algo. Como ejemplos de estos verbos se tiene *salir*, *dormir*, *suicidar*.

c. **Verbos bivalentes**. Los verbos que requieren necesariamente la presencia de dos actantes, la ausencia del segundo actante representa una mutilación del significado del verbo.

d. **Verbos trivalentes.** Verbos que requieren la presencia de tres actantes, como el caso de los verbos *poner* y *dar*.

En la Figura 1 se muestra un ejemplo para cada una de las cuatro categorías de verbos descritas anteriormente. De esta manera, la Figura 1(a) muestra el ejemplo del verbo *llover* con el esquema actancial avalante (sin actores), la Figura 1(b) expone el ejemplo para el verbo *dormir* en el esquema actancial monovalente (con un actor), la Figura 1(c) muestra el ejemplo de un esquema actancial bivalente (con dos actores) mediante el verbo *comer*, finalmente, la Figura 1(d) expone un ejemplo del esquema actancial trivalente (con tres actores) usando el verbo *dar*.

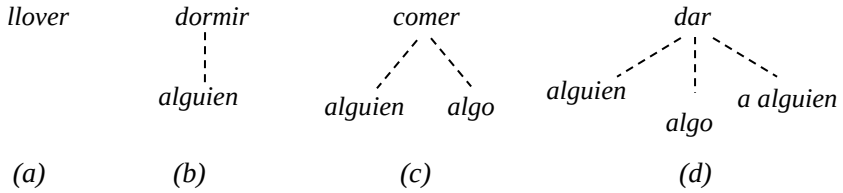

Fig. 1. Ejemplo del uso de verbos en cada categoría: avalante, monovalente, bivalente y trivalente.

La idea de los eventos sobre seguridad como un núcleo forma parte de las bases de este artículo, ya que ayudan a construir un modelo de representación durante la comprensión de los textos y la adquisición de conocimiento. Por ello, estas teorías de eventos otorgan el conocimiento fundamental para la creación del sistema de ontologías de eventos relacionados con la seguridad. Las ontologías individuales presentadas en la Sección 4 se basan en esta cognición de eventos.

4. Creación del sistema de ontologías con bases lingüísticas

En esta sección se presenta el diseño y la creación del sistema de ontologías para la representación de eventos. En este proceso se considera el aspecto de la modularización, el cual consiste, primero, en crear ontologías individuales y, después, integrarlas con la finalidad de construir el sistema de ontologías. Por su parte, la creación de ontologías individuales considera el aspecto de la reutilización, es decir, buscar ontologías disponibles que resuelvan el problema particular.

En nuestro caso, el sistema de ontologías para la representación de eventos sobre seguridad se compone de tres módulos (ontologías individuales): ontología de tiempo,

ontología de espacio y ontología de eventos sobre seguridad, la cual se comporta como el núcleo del sistema de representación.

La ontología de tiempo y la ontología de espacio son ontologías reutilizadas y adaptadas de la literatura, mientras que, la ontología núcleo sobre eventos de seguridad es diseñada y creada desde cero.

En el proceso de creación de la ontología núcleo se consideran aspectos lingüísticos y las características sintácticas de los eventos sobre seguridad con el apoyo de la presencia de este tipo de eventos en textos de noticias.

Se utiliza la sintaxis de Manchester para OWL 1.1 [15] con el propósito de presentar las ontologías individuales y, posteriormente, el sistema de ontologías para la representación semántica de conocimiento en una sintaxis amigable para el usuario. La creación de las tres ontologías individuales se describe a continuación.

4.1. Ontología de tiempo

Los eventos tienen, entre sus características o complementos, al tiempo, que responde a la pregunta ¿cuándo sucedió el evento? Para considerar esta característica en los eventos se ha propuesto la reutilización y adaptación de la ontología llamada *TimeOntology* [16]. Esta ontología individual es adaptada al español para su integración, más adelante, en el sistema de ontologías.

La ontología adaptada tiene las siguientes clases.

```
Class: EntidadTemporal
Class: Intervalo
  SubClassOf: EntidadTemporal
Class: Instante
  SubClassOf: EntidadTemporal
```

La clase *Intervalo* se utiliza para representar una entidad temporal con una extensión o duración. Por su parte, la clase *Instante* es utilizada para representar una entidad temporal con una porción breve de tiempo.

Esta ontología individual será integrada con la ontología de eventos sobre seguridad a través de una relación que será descrita más adelante.

4.2. Ontología de espacio

El espacio es una característica primordial de los eventos, desde que éste expresa el lugar donde se realiza el suceso, respondiendo a la pregunta ¿dónde sucedió? En este rubro se ha adaptado una ontología espacial como módulo (ontología individual) para el sistema de ontologías propuestas en este artículo. Así pues, se ha adecuado la ontología llamada *OntoEspacio* [17], la cual considera la clase *Espacio* como núcleo y a partir de ella, surgen subclases o tipos de espacios.

```
Class: Espacio
Class: EspacioUrbano
  SubClassOf: Espacio
```

```
Class: EspacioGeográfico
  SubClassOf: Espacio
Class: EspacioPúblico
  SubClassOf: EspacioUrbano
Class: EspacioPrivado
  SubClassOf: EspacioUrbano
Class: Ecosistema
  SubClassOf: EspacioGeográfico
Class: CoordandasGeográficas
  SubClassOf: EspacioGeográfico
```

Esta ontología individual será integrada con la ontología de eventos sobre seguridad a través de una relación que será descrita más adelante.

4.3. Ontología núcleo de eventos sobre seguridad

La ontología núcleo considera a los eventos sobre seguridad como su parte primordial. Es por ello que esta ontología se basa en la clase llamada *EventoSeguridad* de la cual se desprenden cuatro subclases, es decir, en este trabajo se consideran cuatro tipos de eventos sobre seguridad.

```
Class: Suicidio
  SubClassOf: EventoSeguridad
Class: Ataque
  SubClassOf: EventoSeguridad
Class: Arresto
  SubClassOf: EventoSeguridad
Class: Hallazgo
  SubClassOf: EventoSeguridad
```

Adicionalmente se define un conjunto de clases para representar los actantes de la teoría de *Tesnière* [13]. A partir de dichas teorías se obtienen las clases Actor y Objeto, de las cuales se desprenden las siguientes subclases.

```
Class: Organización
  SubClassOf: Actor
Class: Persona
  SubClassOf: Actor
Class: ObjetoDescubierto
  SubClassOf: Objeto
Class: ObjetoAtaque
  SubClassOf: Objeto
```

Los cuatro tipos de eventos sobre seguridad y los actantes tienen una serie de relaciones semánticas o relaciones ontológicas. Estas relaciones están basadas en la teoría de eventos y las valencias de los verbos presentadas anteriormente.

Con estas bases lingüísticas se caracteriza al evento *Suicidio* como un evento o verbo monovalante, lo cual genera una sola relación entre el evento y un actor.

```
ObjectProperty: esCometidoPor
    Domain: Suicidio
    Range: Persona
```

En la Figura 2(a) se puede observar el esquema actancial del evento *Suicidio* y en la Figura 2(b) se presenta un ejemplo de un evento de este tipo.

Fig. 2. Esquema actancial monovalente del evento *Suicidio*.

Por su parte, el evento *Ataque* se caracteriza como un evento o verbo trivalente en el cual intervienen el atacante (persona que realiza el ataque), el atacado (persona lastimada en el ataque) y el instrumento (objeto o herramienta con la cual fue realizado el ataque). Por lo tanto, tres relaciones son generadas a partir de este tipo de evento.

```
ObjectProperty: tieneAtacante
    Domain: Ataque
    Range: Persona OR Organización
ObjectProperty: tieneInstrumentoDeAtaque
    Domain: Ataque
    Range: ObjetoAtaque
ObjectProperty: tieneLastimado
    Domain: Ataque
    Range: Persona OR Organización
```

En la Figura 3(a) se puede observar el esquema actancial del evento *Ataque* y en la Figura 3(b) se presenta un ejemplo de este tipo de evento.

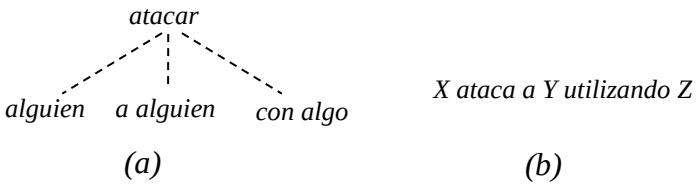

Fig. 3. Esquema actancial trivalente del evento *Ataque*.

Con respecto al evento de tipo *Hallazgo* o *Descubrimiento* se han caracterizado dos relaciones entre el evento y los actantes. De esta manera, se tiene que este tipo de evento

es considerado con un valor actancial bivalente, lo cual origina las siguientes relaciones en el modelo ontológico.

```
ObjectProperty: hallazgoRealizadoPor
    Domain: Hallazgo
    Range: Organización
ObjectProperty: tieneObjeto
    Domain: Hallazgo
    Range: ObjetoDescubierto
```

En la Figura 4(a) se puede observar el esquema actancial del evento *Hallazgo* con una relación hacia la clase *Organización* (quien realiza el hallazgo o descubrimiento) y una segunda relación hacia un *Objeto* (elemento encontrado). Por su parte, en la Figura 4(b) se presenta un ejemplo de esta categoría de eventos.

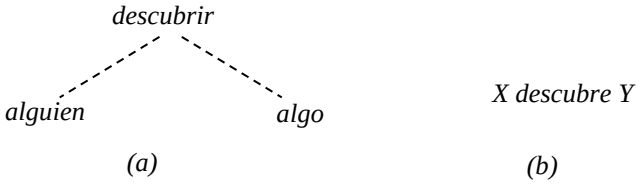

Fig. 4. Esquema actancial bivalente del evento *Descubrimiento*.

El evento de tipo *Arresto* se ha caracterizado con un esquema actancial bivalente. Por ello, se representan dos relaciones semánticas, una con la clase *Organización* (quien realiza el arresto) y otra relación para indicar al arrestado (persona que es detenida). Esto genera el siguiente código en la ontología.

```
ObjectProperty: arrestoRealizadoPor
    Domain: Arresto
    Range: Organización
ObjectProperty: tienePersonaArrestada
    Domain: Arresto
    Range: Persona
```

En la Figura 5(a) se puede observar el esquema actancial del evento *Arresto* y en la Figura 5(b) se presenta un ejemplo de esta clase de eventos.

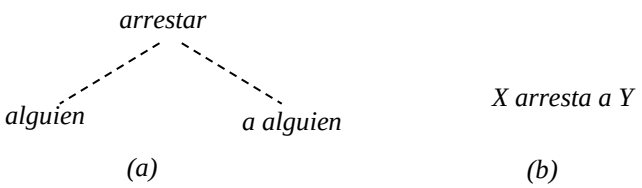

Fig. 5. Esquema actancial bivalente del evento *Arresto*.

4.4. Integración de ontologías individuales

El proceso de integración corresponde a la creación de relaciones semánticas entre las ontologías individuales, ya sea que fueron creadas desde cero o adaptadas de ontologías existentes. El objetivo de este proceso es proveer una solución integral, con un sistema de ontologías, a la representación de eventos sobre seguridad considerando sus dimensiones de tiempo, espacio y actores.

Esta integración da origen a la relación llamada *sucedeEn*, la cual se utiliza para representar el lugar donde sucede el evento sobre seguridad. Esta relación se crea en el sistema de ontologías de la siguiente manera.

```
ObjectProperty: sucedeEn
   Domain: EventoSeguridad
   Range: Espacio
```

Adicionalmente se crea la relación *tieneTiempo*, la cual se utiliza para responder a la pregunta ¿cuándo sucedió el evento sobre seguridad? Esta relación se crea en el sistema de ontologías.

```
ObjectProperty: tieneTiempo
    Domain: EventoSeguridad
    Range: EntidadTemporal
```

De esta manera quedan integradas las tres ontologías individuales (ontología de espacio, ontología de tiempo y ontología de eventos) con la finalidad de crear un sistema de ontologías con relaciones semánticas entre ellas. El esquema general del sistema de ontologías se puede observar en la Figura 6.

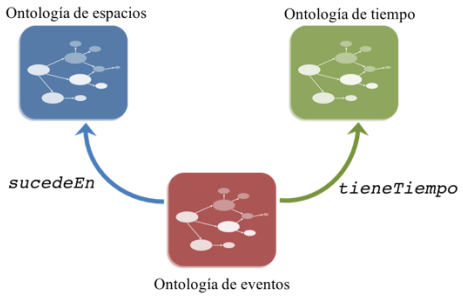

Fig. 6. Integración entre las ontologías individuales.

5. Axiomatización del sistema de ontologías

El sistema de ontologías creado con sus características para cada clase es instanciado con miembros o |concepto con la finalidad de validar la consistencia de la ontología. Diferentes tipos de axiomas son considerados en el sistema de ontologías para la representación de eventos sobre seguridad, los cuales se describen a continuación.

5.1. Axiomas de definición de clases

Los axiomas de definición de clases se utilizan para restringir los miembros pertenecientes a una clase, es decir, determinar si dos clases tienen elementos disjuntos o no. En nuestro sistema de ontologías, las siguientes restricciones de definición de clases son creadas.

```
Class: Suicidio
DisjointWith:
  Arresto, Ataque, Hallazgo

Class: Organización
DisjointWith:
  Persona, ObjetoDescubierto, ObjetoAtaque
```

5.2. Axiomas de definición de propiedades de datos y objetos

Este tipo de axiomas define restricciones de tipo cardinalidad, existencial, universal y de valor para las características de las clases. Por otro lado, también, define restricciones para establecer que ciertos individuos de una clase tienen relación con individuos de otra clase específica. En este aspecto, se definen una serie de axiomas de restricción para nuestro sistema de ontologías.

```
Suicidio esCometidoPor some Person

Ataque tieneAtacante some (Person OR Organization)
Ataque tieneInstrumentoDeAtaque exactly 1 ObjetoAtaque
Ataque tieneLastimado some (Person OR Organization)

Hallazgo hallazgoRealizadoPor some Organización
Hallazgo tieneObjeto some ObjetoDescubierto

Arresto arrestoRealizadoPor exactly 1 Organización
Arresto personaArrestada some Organización

EventoSeguridad sucedeEn exactly 1 Espacio
EventoSeguridad tieneTiempo exactly 1 EntidadTemporal
```

5.3. Axiomas de poblado de individuos

El poblado del sistema de ontologías consiste en crear un conjunto de individuos para las clases con la finalidad de hacer posible la evaluación de los axiomas descritos anteriormente. Además, este tipo de axiomas permiten la verificación de la consistencia del sistema de ontologías ya que permiten validar las restricciones de clases, propiedades y objetos. En este aspecto, los siguientes hechos, como axiomas representativos,

son agregados al sistema ontológico, para los cuales, no se agregan nombres de personas en su lugar se incluyen nombres de instancias anónimas.

```
Individual: personaX
  Types:
    Persona

Individual: unSuicidio
  Types:
    Suicidio
  Facts:
    esCometidoPor personaX
    sucedeEn 'Guadalajara, Jalisco'
    tieneTiempo '2 de octubre'
```

6. Conclusiones y trabajo futuro

En este artículo se ha presentado la construcción de un sistema de ontologías para la representación de eventos relacionados con la seguridad. La construcción del sistema de ontologías se ha llevado a cabo siguiendo el principio de modularidad, el cual ha consistido en reutilizar, adaptar y crear tres ontologías individuales. Por un lado, se reutilizan y adaptan de la literatura una ontología de espacio y una ontología de tiempo. Por su parte, la ontología individual sobre eventos de seguridad es creada desde cero considerando la teoría de eventos de [13] con sus características.

Las ontologías individuales creadas son integradas para formar el sistema ontológico general, de esta manera, se obtiene un modelo ontológico para la representación de eventos sobre seguridad, donde se puede representar, adicionalmente, su espacio (lugar donde sucedió) y el tiempo.

La aportación principal de este artículo se centra en lograr la creación de un sistema ontológico, basado en la modularidad, para la representación de eventos relacionados con seguridad. Este sistema puede ser de gran utilidad para apoyar diversas aplicaciones como sistemas de pregunta-respuesta o buscadores semánticos de información sobre eventos.

Como trabajo futuro, se plantea el poblado semiautomático del modelo ontológico de eventos sobre seguridad a partir de noticias de periódicos en español usando patrones lingüísticos, los cuales se podrían obtener de las relaciones semánticas en la ontología. Así como también técnicas de reconocimiento de entidades nombradas para identificar y clasificar las entidades de personas, objetos y organizaciones. Enriquecer las relaciones con sinónimos representa una tarea a futuro que puede ayudar a mejorar la tarea de descubrimiento de eventos a partir de textos.

Referencias

1. Weigand, H.: Multilingual ontology-based lexicon for news filtering. In: K. Mahesh, editor, The TREVI Project, 138–159 (1997)
2. Bravo, M., Reyes-Ortiz, J. A., Alcántara, R., Rodríguez, J.: Addresing Clarity and Coherence during Ontology Construcction. In: Proceedings of the Mexican International Conference on Computer Science (2016)
3. Piskorski, J., Tanev, H., Wennerberg, P. O.: Extracting violent events from on-line news for ontology population. In: International Conference on Business Information Systems, 287–300 (2007)
4. Obrst, L., Chase, P., Markeloff, R.: Developing an Ontology of the Cyber Security Domain. In: The 11th International Conference on Semantic Technology for Intelligence, Defense, and Security, 49–56 (2012)
5. Duncan, J., Eilbeck, K., Narus, S. P., Clyde, S., Thornton, S., Staes, C.: Building an Ontology for Identity Resolution in Healthcare and Public Health. Online journal of public health informatics 7(2) (2015)
6. Vargas-Vera, M., Celjuska, D.: Event recognition on news stories and semi-automatic population of an ontology. In: Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence, 615–618 (2004)
7. Borsje, J., Hogenboom, F., Frasincar, F.: Semi-automatic financial events discovery based on lexico-semantic patterns. International Journal of Web Engineering and Technology, 6(2), 115–140 (2010)
8. Adam, T. J., Wang, J.: Adverse Drug Event Ontology: Gap Analysis for Clinical Surveillance Application. In: AMIA Summits on Translational Science Proceedings, 16–20 (2015)
9. Gurulingappa, H., Mateen-Rajpu, A., Toldo, L.: Extraction of potential adverse drug events from medical case reports. Journal of biomedical semantics, 3, 15 (2012)
10. He, Y., Sarntivijai, S., Lin, Y., Xiang, Z., Guo, A., Zhang, S., Smith, B.: OAE: the ontology of adverse events. Journal of biomedical semantics, 5, 29 (2014)
11. Winnenburg, R., Shah, N. H.: Generalized enrichment analysis improves the detection of adverse drug events from the biomedical literature. BMC bioinformatics, 17, 250 (2016)
12. Miller, G., Johnson-Laird, P.: Language and Perception. Belknap Press, Cambridge, UK (1976)
13. L. Tesnière. Eléments de syntaxe structural. second edition, Librairie Klincksieck, Paris (1976)
14. Halliday, M.: An introduction to functional grammar. 2nd edition, Edward Arnold, London (1994)
15. Horridge, M., Patel-Schneider, P. F.: Manchester syntax for OWL 1.1, OWL: Experiences and Directions. Washington (2008)
16. Hobbs, J. R., Pan, F.: Time ontology in OWL. W3C working draft, 27, 133 (2006) `https://www.w3.org/TR/owl-time/`
17. Gómez, J. D.: Poblado de ontologías espaciales a partir de texto no estructurado. Tesis de Maestría, Centro Nacional de Investigación y Desarrollo Tecnológico, Morelos, México (2012)

ISSN 1870-4069

Recuperación, procesamiento y clasificación de tuits para visualizar estructuras de interacción

Carlos Pérez[1], Jorge Cortés[1], Aarón Ramírez[1], Rocío Abascal-Mena[1], Alejandro Molina-Villegas[2]

[1] Universidad Autónoma Metropolitana-Cuajimalpa, México

[2] Comisión Nacional para el Conocimiento y Uso de la Biodiversidad, México

{2143805174, 2143805218, 2113066212}@alumnos.cua.uam.mx
mabascal@correo.cua.uam.mx, amolina@conabio.gob.mx

Resumen. En un contexto de medios sociales digitales, donde existen múltiples formas de vinculación entre usuarios, resulta importante contar con herramientas que permitan analizar los procesos de interacción presentes en estas plataformas. El análisis de redes sociales utiliza frecuentemente diagramas nodo-enlace para representar las relaciones entre un conjunto de actores. Sin embargo, la representación visual de grafos con información adicional en vértices y aristas es una tarea compleja y pocos programas cuentan con esta característica. Presentamos una propuesta para la recuperación y procesamiento de tuits con el fin de visualizar redes de Comunicación Política en Twitter. El sistema incluye clasificación automática de usuarios, diferenciación del tipo de aristas dependiendo de si es una mención, una respuesta o un retuit, así como visualización interactiva de los grafos.

Palabras clave: Visualización de redes complejas, interacción en medios sociales, clasificación automática, ciencias de datos en comunicación política.

Recovery, Processing and Classification of Tweets to Visualize Structures of Interaction

Abstract. In the context of digital social media, where users have multiple ways to interact with others, it is important to have tools to analyze the interaction processes within these platforms. Social network analysis frequently uses node-link diagrams to represent relationships among social actors. However, the visual representation of network graphs with additional information in vertices and edges is a complex task and few programs provide this feature. We propose a system for the recovery and processing of tweets to visualize Political Communication networks. The

system includes the automatic classification of Twitter users, differentiation between retweets, mentions, and replies, as well as an interactive visualization of network graphs.

Keywords. Visualization of complex networks, interactions on social media, automated classification, data science in political communication.

1. Introducción

Los medios sociales se han instalado progresivamente en nuestra vida diaria, alterando los métodos de comunicación y los intercambios de información. Estas plataformas sociales continúan evolucionando y permitiendo nuevas formas de acción colectiva. A partir de ello, es posible identificar una nueva Comunicación Política apoyada en el uso de medios sociales. Los actores sociales trascienden de ser consumidores hacia productores creativos de información sustantiva, llamados prosumidores.

Twitter, con sus no más de 140 caracteres, se ha convertido en una herramienta de manifestación social y política en la que los prosumidores no sólo crean mensajes sino que reproducen y responden creando un medio de colaboración. Es así como el análisis de datos, recuperados de Twitter, puede proporcionar un medio para observar la sociedad contemporánea. A partir de los intereses, motivaciones y actitudes de los usuarios, es posible descubrir patrones de comportamiento [6]. Los tuiteros tienen a su disposición diversas formas de interacción, como las menciones, las respuestas, los retuits y los *likes*. Esta variedad presenta retos para el análisis de las interacciones en Twitter, especialmente cuando se asigna algún significado a cada uno de los tipos de interacción en el contexto de investigaciones específicas. Una forma frecuente de examinar las interacciones presentes en los medios sociales es generar grafos que representan relaciones entre actores con aristas (o enlaces) y vértices (o nodos), respectivamente. Sin embargo, no todos los programas especializados en análisis de redes permiten dibujar múltiples aristas entre un mismo conjunto de nodos. La última versión de Gephi (http://gephi.org) los soporta, pero únicamente en su laboratorio de datos y no en la representación gráfica; por su parte Social Network Visualizer (http://socnetv.sourceforge.net/) sí los muestra, pero de manera separada, es decir, dibuja un grafo por cada tipo de interacción.

Asimismo, hay complicaciones relacionadas con la recuperación y el filtrado de los datos, pues el software especializado suele restringir la personalización de la salida de datos recuperados. Los formatos empleados para su almacenamiento, al no estar destinados para su manipulación a través de distintos programas, pueden ocasionar pérdidas de información y, además, limitar su compatibilidad con herramientas de visualización. Una ausencia de capacidades para el tratamiento del corpus reduce las opciones para la depuración y extracción de características representativas de los conjuntos de datos.

Por todo lo anterior, en este artículo se proponen técnicas de recuperación, depuración y procesamiento de datos de Twitter para la visualización y análisis de datos relacionales complejos. La visualización se utiliza en el estudio de las interacciones entre ciudadanos, políticos y medios de noticias en Twitter, con el objetivo de conocer de qué manera interactúan dentro de este medio social.

El artículo está organizado de la siguiente manera: en la Sección 2 se presenta el estado del arte de los principales trabajos en el análisis de Twitter. La propuesta de recuperación, procesamiento y visualización del conjunto de datos es detallada en la Sección 3. Un análisis de los resultados es presentado en la Sección 4. Finalmente, se da un panorama general sobre el estado actual del trabajo y lo que se espera en un futuro.

2. Estado del arte

Desde el surgimiento de la Web 2.0, el modelo de Comunicación Política tradicional se ha redefinido al permitir un acercamiento de la ciudadanía con los políticos y los medios. Es así como el llamado régimen mediático se ha visto opacado con la participación de los ciudadanos en las redes sociales, quienes interactúan entre ellos creando, retransmitiendo e interactuado con otros actores. Esto amplía el espacio de opinión pública y, potencialmente, podría mejorar las condiciones democráticas a partir de una mayor participación y representación de la ciudadanía [12].

Son numerosos los estudios que se han centrado en el estudio de Twitter. En específico, se encuentran trabajos sobre la detección de actores influyentes ([4,10,14,18]), el desarrollo de campañas políticas en Internet, tanto en México ([3]) como en otros paises ([5,7,9,10]) y la predicción y análisis de los usuarios de acuerdo con lo que comparten y sus principales contactos como es el caso de ([1,11,17]).

Sin embargo, en los estudios antes mencionados, el análisis está centrado en el contenido y sentimiento expresado en los tuits o en la actividad de personajes específicos, no en las interacciones entre ellos.

Consideramos que las interacciones es un aspecto importante a estudiar ya que como se menciona en [15], las interacciones de los actores de no élite, nombrados por Chadwick [2], tienen mucho éxito al utilizar redes sociales como Twitter. Sabiendo que en cuanto más los medios tradicionales realicen difusión en los medios digitales será más probable que los ciudadanos activos, que utilizan las mismas herramientas, puedan influir en la cobertura de los medios de comunicación. De igual forma, los actores de no élite tienen cuidado en su interacción con las élites en línea, incluyendo políticos y periodistas, haciendo pues que la interacción tenga un papel importante en la cobertura de las noticias.

La propuesta está compuesta de 3 pasos que incluyen: 1) recuperación de tuits, 2) procesamiento y 3) visualización. La recuperación o minado de tuits es, generalmente, realizado a partir del uso de *hashtags* asociados a los tuits. Sin embargo, la detección del ruido causado por *hashtags* que no tienen relación alguna con el contexto es una tarea ardua. En general, las investigaciones lo

afrontan como un problema de clasificación proponiendo, en algunos trabajos, un enfoque supervisado basado en grafos con el fin de inferir las categorías de intención de los tuits ([6,13,19]).

Por su parte, la recuperación de tuits tiene como objetivo el filtrado mediante un análisis del tuit. Este procesamiento incluye la limpieza de los tuits a partir de una comparación con términos o *stopwords*. El filtrado, y en algunos casos jerarquización, ha sido muy estudiado para los casos de documentos estáticos. Sin embargo, en las redes sociales existen nuevos factores que lo vuelven difícil como lo es el uso de idiomas distintos ([3]), estilos fragmentados de redacción, la ambigüedad, y la restricción propia de los 140 caracteres como en el caso de Twitter. Algunos métodos están basados en modelos probabilísticos o clasificadores como Naïve Bayes. Sin embargo, la gran parte de los trabajos encontrados son aplicados en el análisis de sentimientos en cuyo caso es muy importante el análisis sintático y semántico de todo el tuit. No hay ejemplos de aplicaciones en el que se pueda estudiar la interacción de los actores sin tomar en cuenta el contenido del tuit.

A la fecha, el impacto de Twitter en la Comunicación Política, en particular en el contexto mexicano, no ha sido abordado suficientemente aún para ofrecer un panorama claro sobre las dinámicas entre actores dentro del medio. Es de gran importancia estudiar el tipo de interacciones que se dan entre los diferentes actores ya que, como lo menciona [15], los modelos de toma de decisión periodística y de Comunicación Política necesitan incorporar el papel de las plataformas en los medios sociales, como Twitter, debido a su gran importancia.

En la siguiente sección se presenta la propuesta para la recuperación, procesamiento y visualización de estructuras de interacción.

3. Propuesta de recuperación, procesamiento y visualización

A grandes rasgos, nuestra propuesta para obtener una visualización de conjuntos de datos para análisis tiene tres etapas principales: la Recuperación, el Procesamiento y la Visualización. Entre las características principales de este flujo, destacan que las primeras dos etapas consideran una salida de datos para su posterior visualización en un sistema desarrollado a medida. De igual manera, destaca la inclusión de procesamiento en paralelo para la generación de un modelo de clasificación automática de perfiles de usuario. La Figura 1 detalla el proceso.

3.1. Recuperación

Se automatizó el proceso de recuperación de información mediante un *script* para la captura de publicaciones mediante la API de Twitter empleando la librería Tweepy (http://www.tweepy.org/). Entre otros parámetros, se limitó la recuperación al idioma español y conteniendo los términos especificados incluidos en los metadatos determinados por la plataforma.

Como detalla la Figura 2, tras la obtención de cada tuit, se realizó una búsqueda de dos niveles de respuestas para obtener entradas relacionadas con la temática recuperada que no incluyeran necesariamente los términos inicialmente establecidos.

Uno de los principales problemas de la recuperación consistió en que al realizar peticiones repetidas se obtenían datos duplicados. Por lo tanto, se empleó el metadato *id* de cada publicación y se consultó su existencia en el conjunto de entradas recuperadas. Para evitar una disminución en el rendimiento del programa, debido al gran número de tuits almacenados, los archivos se guardaron en texto plano. Los documentos fueron separados por día de publicación –obtenido del metadato *created_at*– para reducir el tiempo de cómputo.

3.2. Depuración

Los datos recuperados fueron procesados con el fin de conservar publicaciones relevantes para su análisis. En la Figura 3 se muestra la serie de filtros ordenados, cuyos resultados fungieron como entrada del paso subsecuente para reducir el número de entradas.

El primer paso del proceso de depuración implicó descartar los tuits sin interacción y, por lo tanto, ninguna conexión con otros actores.

Definimos que un tuit tiene interacción si cumple con alguno de los siguientes criterios:

- es un retuit;
- menciona a otro usuario o;
- es una respuesta a un tuit previo.

Mediante estos criterios se conservaron aquellos tuits que no estaban aislados y eran susceptibles a representarse mediante un grafo.

Es posible encontrar tuits publicados por métodos automáticos conocidos como *bots* y cuyo propósito es popularizar o desprestigiar a una persona o un determinado tema. Una característica común en los *bots* reside en su plataforma

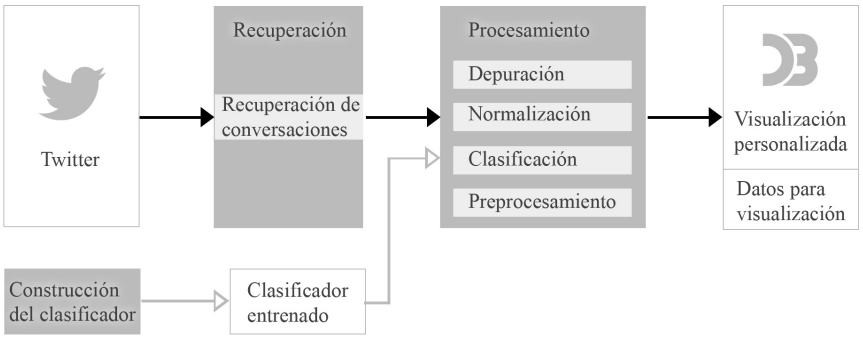

Fig. 1. Flujo de recuperación, procesamiento y visualización.

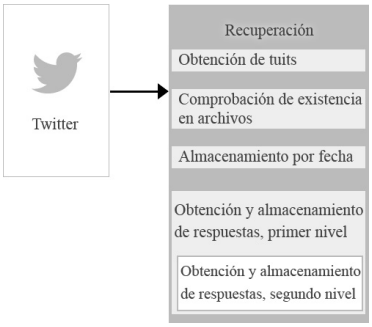

Fig. 2. Flujo de recuperación de publicaciones.

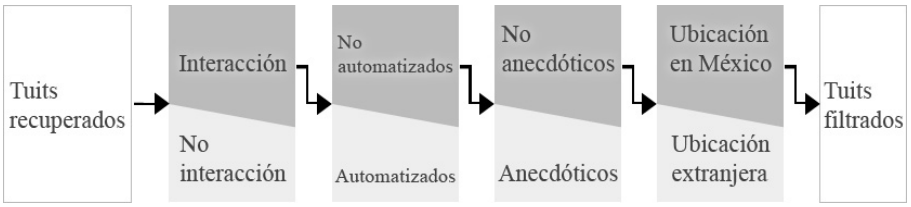

Fig. 3. Flujo de depuración de entradas.

de publicación, diferente a las oficiales creadas por Twitter. Algunas de estas plataformas son IFTTT (https://ifttt.com/) y Tapbots (http://tapbots.com/). En nuestra propuesta, el módulo de depuración excluye este subconjunto, conservando así únicamente, los tuits presumiblemente auténticos. Cabe mencionar que aún cuando esta estrategia resultó suficiente para nuestros experimentos, no existe actualmente un método infalible para filtrar *bots*.

Se incluyó un filtro de tuits que tratasen de una experiencia anecdótica o contenido promocional. Con este fin, se creó una lista de 43 términos para detectar y excluir estas ocurrencias. Sólo se mantuvieron los tuits que no tenían ocurrencia alguna de los términos establecidos en la lista.

Finalmente, se descartaron los tuits con una ubicación geográfica fuera de México. Para este proceso se usó una lista de localidades de la República Mexicana y se conservaron los tuits emitidos desde alguna de las ubicaciones listadas y aquellos sin ubicación.

Al concluir el proceso de depuración, prevalecieron los tuits que tenían interacción, que no fueron publicados por *bots*, que no eran de carácter anecdótico o promocional y que fueron emitidos desde México.

3.3. Normalización y clasificación

Las cuentas presentes en el corpus depurado fueron clasificadas en tres categorías: medio, político y ciudadano. Este proceso de clasificación fue realizado manualmente en primera instancia y luego automáticamente mediante algorit-

mos de aprendizaje supervisado. En la Figura 4 se detallan los componentes del proceso para el entrenamiento del clasificador.

Se usó la descripción del perfil de cada usuario como criterio para determinar la clase de cada cuenta, así como su rol en Twitter a partir de esta información.

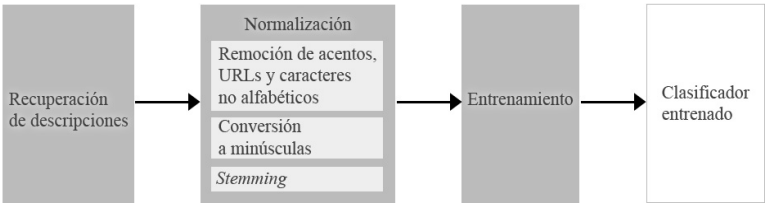

Fig. 4. Flujo de generación del clasificador bayesiano.

Primeramente se clasificó de manera manual un subconjunto de perfiles pertenecientes a las categorías medio y político. Para este proceso, se automatizó la descarga de sus descripciones de perfil (biografías). Seguidamente, se empleó un *script* para remover hipervínculos, signos de puntuación y caracteres especiales, con la finalidad de conservar sólo caracteres alfanuméricos. El texto restante de cada descripción fue transformado a minúsculas y cada palabra reducida a su raíz léxica (*stem*) usando la implementación del algoritmo de Porter contenido en el módulo *Snowball Stemmer* de la biblioteca *NLTK*.

Como resultado del procesamiento descrito anteriormente, se obtuvo la representación de *Bolsa de Palabras* de un subconjunto de descripciones de perfiles de usuarios de medios y políticos. A este proceso se le conoce como codificación y sirve para transformar texto en vectores numéricos que la máquina pueda procesar. La manera precisa de codificar la información depende de cada problema particular pero lo que es indispensable para este tipo de métodos es transformar la información textual en datos vectoriales que son la materia prima del aprendizaje supervisado.

La clasificación automática de perfiles de usuario se realizó mediante técnicas de reconocimiento de patrones basado en el teorema de Bayes.

Básicamente, el teorema de Bayes (ver Ecuación 1), define una manera de calcular probabilidades condicionales. Por ejemplo, si $P(mediatico)$ es la probabilidad a priori de que un perfil de usuario sea un perfil de medios. $P(mediatico|x)$ sería la probabilidad a posteriori, de ser un perfil de medios, basada en el contenido del perfil, siendo x nuestra nueva observación, es decir, la codificación de un perfil de usuario de Twitter en forma de vector:

$$P(mediatico|x) = \frac{P(x|mediatico)P(mediatico)}{P(x)}. \quad (1)$$

Así, para aplicar el teorema de Bayes al reconocimiento de patrones, se define $p(x)$ como la probabilidad de que exista una codificación como la de la entrada x. Usando este marco metodológico, solamente hace falta conocer, a través de

muchos ejemplos, cómo se comportan los elementos de una clase particular (descripciones de perfiles mediáticos, o sea la variable *mediatico*), y de ésta forma hallar la función de distribución asociada a esa clase, que será asímismo $p(x|mediatico)$. Lo que nos permite saber, a partir de la probabilidad de una clase, la probabilidad de dicha clase una vez obtenido el patrón x. En concreto, de entre todas las clases posibles, debemos escoger la de mayor probabilidad a posteriori.

Los datos de entrenamiento ingresados al clasificador de nuestro sistema fueron las cuentas de usuarios de medios y políticos representados por sus descripciones previamente clasificadas por inspección.

De este modo, el clasificador entrenado es empleado como parte del sistema para automatizar la categorización de nuevas cuentas. Aunque la precisión de dicho modelo no es perfecta, es lo suficientemente buena para ser usada en el sistema en producción. La Figura 5 muestra en detalle el número de instancias de perfiles clasificados como políticos que en efecto sí son políticos (verdaderos positivos para p). Análogamente, se muestran los verdaderos positivos de perfiles mediáticos (verdaderos positivos para m); así como los correspondientes falsos positivos para p y falsos positivos para m. Se puede deducir, a partir de esta matriz, que el 95.6% de perfiles es clasificado correctamente. En la sección 4. se muestran más detalladamente los resultados de la clasificación automática de perfiles de usuario.

		Clasificación automática	
		p	m
Real	p	1183	59
	m	40	1001

Fig. 5. Matriz de confusión resultante de la clasificación automática de perfiles de usuarios de twitter p= perfiles políticos; m=perfiles mediáticos.

3.4. Visualización

Los diagramas nodo-enlace son empleados frecuentemente para representar visualmente datos relacionales, pues a partir de ellos es posible obtener una idea general de los patrones de actividad dentro de una red. Debido a ello, se decidió emplearlos para visualizar las interacciones entre los usuarios de Twitter.

El primer paso fue establecer parámetros visuales para los nodos y los enlaces. Cada nodo representa una cuenta de Twitter y su color indica la pertenencia a una de las tres categorías. Siguiendo las recomendaciones de [16], se utilizaron

tonos notoriamente distintos entre sí: magenta, cian y gris. De igual forma, se asignaron tres colores para diferenciar los tres tipos de interacción. El fondo sobre el que se dibujarían las redes debía procurar un buen contraste para diferenciar los tonos, así que se optó por un color oscuro.

El tamaño de los elementos también representa una dimensión de los datos. En el caso de los nodos, el área se calcula con base en el número de enlaces recibidos (*in-degree*) o bien de los emitidos (*out-degree*). De esta manera, se puede identificar a las cuentas más solicitadas o las más activas. En cuanto a los enlaces, el grosor de las líneas aumenta según el número de interacciones entre dos cuentas.

La principal característica de nuestras redes es la multiplicidad de formas de interacción entre las cuentas. Para observar los tres tipos de comunicación, cada uno se dibuja con un enlace y, de existir más de un tipo de vinculación entre dos cuentas, se añade otra línea y se modifica su curvatura para diferenciarla de la primera. Así, pueden haber hasta seis enlaces entre dos nodos A y B: tres de A hacia B y viceversa.

Se decidió trabajar con D3 –una librería de JavaScript para generar y manipular documentos web con datos–, debido a que soporta un amplio número de representaciones gráficas y brinda gran control sobre los atributos visuales y la interactividad. Para obtener grafos más legibles se modificaron los atributos predeterminados del algoritmo de fuerza de D3. Se redujo la gravedad y se aumentó la longitud de las aristas con el fin de dispersar el grafo y observar mejor las interacciones. En la Figura 6 se muestra una de las redes obtenidas.

3.5. Interactividad

Sintetizar en una imagen el conjunto de datos presentes en redes multivariantes es una tarea desafiante y, en ocasiones, resulta imposible mostrar todos los datos de manera útil [8]. En los diagramas nodo-enlace, surgen problemas de legibilidad a medida que el volumen de datos aumenta. La posibilidad de interactuar con la representación es un aspecto esencial para obtener información de la visualización. La inclusión de funciones como *zoom*, *panning*, resaltado, filtrado o búsqueda permite a los usuarios ubicar zonas y actores de interés [20].

En la presente propuesta de visualización, se incluyen las funciones de zoom y arrastre para acercar, alejar y desplazar el grafo. El usuario puede ocultar los enlaces para observar solamente la distribución de los nodos en el grafo. En caso de que necesite concentrarse sólo en los enlaces, puede quitar el color a los nodos.

Para ubicar rápidamente las cuentas principales, se colocó un botón que muestra los nombres de los diez nodos con mayor grado de entrada. Se incluyó un campo de búsqueda para localizar cuentas específicas. Al hacer clic en *Buscar*, se resaltan el nodo y sus vecinos. Asimismo, cuando se pasa el cursor sobre un nodo, se muestran su nombre, grado de entrada y grado de salida. Dar clic en el nodo selecciona su red inmediata y muestra la frecuencia de interacción de los enlaces cuyo peso es mayor a uno, porque a simple vista no es posible comparar con precisión su grosor. Un par de radio *buttons* permite alternar entre los valores de grado de entrada o salida para determinar el tamaño de los nodos.

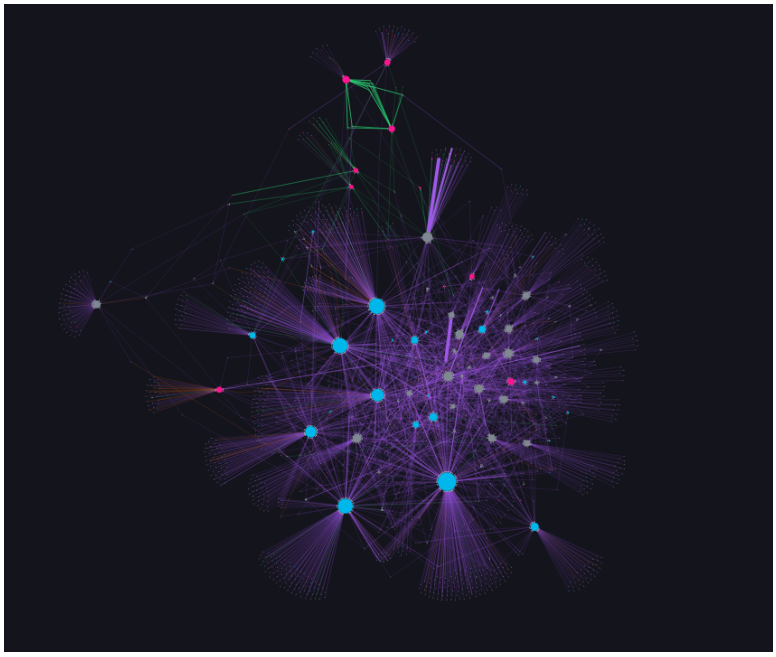

Fig. 6. Red de interacciones.

En cuanto a los filtros, es posible observar cada uno de los tipos de interacción por separado o cualquier combinación de ellos. Otro conjunto de botones filtra los nodos según su clasificación. Además, oculta aquellos nodos relacionados únicamente con cuentas de la clasificación filtrada. Por último, los usuarios pueden observar sólo aquellas relaciones recíprocas entre nodos. En la Figura 7 se muestra una red y en la parte izquierda de la pantalla el menú que contiene las herramientas de filtrado.

4. Resultados

El método propuesto para la obtención y tratamiento de datos, permite una simplificación en el flujo de análisis de grandes cantidades de entradas. La recuperación de cadenas de conversaciones implicó la incorporación de datos relacionados con la temática recuperada que, de otra manera, sería complejo vincular. La depuración automatizada de publicaciones posibilitó la obtención de corpus condensados con una menor variación temática entre sus publicaciones. Al no contar con un corpus anotado ni con un *gold standard* para evaluar el desempeño con los datos actuales, se utilizó una validación cruzada con 10 pliegues en Weka. Los resultados detallados de la evaluación y del clasificador de perfiles son presentados en las Tablas 1 y 2 respectivamente.

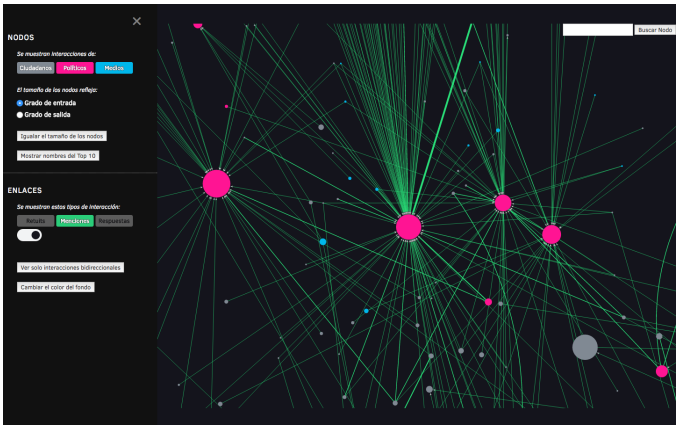

Fig. 7. Controles de la visualización que permiten navegar y filtrar el grafo.

Tabla 1. Resultados de la validación cruzada con 10 pliegues para el modelo de clasificación.

Indicador	**valor**	
Instancias clasificadas correctamente	2184	95.6%
Instancias clasificadas incorrectamente	99	4.3%
Coeficiente kappa	0.9127	
Error promedio absoluto	0.0537	
Error cuadrático Medio	0.1822	
Error absoluto relativo	10.82 %	
Número de instancias en la evaluación	2283	

Así, la clasificación automatizada permite obviar parte de la manipulación de la información recabada.

Pese a los resultados, existe un gran número de instancias incorrectamente categorizadas, debido a las formas de uso de Twitter, en específico la capacidad de los usuarios para ingresar información arbitraria a su perfil, dando lugar a inconsistencias y, por ende, disminuyendo la precisión del sistema de clasificación.

A partir del análisis de las redes generadas, se observa un uso principalmente informativo de Twitter, pues la forma de interacción más empleada es el retuit y se concentra alrededor de cuentas de medios de noticias. La actividad de los usuarios se enfoca en la difusión de notas periodísticas sobre los temas

Tabla 2. Resultados de la clasificación automática de usuarios.

Clase	**Precisión**	**Cobertura**	**F-measure**
Político	0.967	0.952	0.960
Medio	0.944	0.962	0.953
Promedio ponderado	0.957	0.957	0.957

analizados. Un rasgo característico es que la mayoría de los usuarios están vinculados solamente a un medio, siendo muy pocos quienes retuitean contenido de varias fuentes. Las menciones son utilizadas generalmente para dirigir mensajes a ciertos actores o para hablar sobre ellos. Es por ello que existen regiones de color verde en los grafos donde, por lo regular, hay cuentas de políticos. La respuesta es el mecanismo de interacción menos utilizado, lo cual parece indicar poca propensión al diálogo en las redes que se estudiaron.

La presencia de cuentas de políticos y medios es muy escasa. En promedio, el 90% de los nodos fue clasificado como ciudadano. Se observó en los actores políticos y de medios un uso estratégico de Twitter. Por ejemplo, cuentas pertenecientes al mismo partido político retuitean el contenido publicado por algún líder de su organización. Los medios muestran un comportamiento similar, con periodistas difundiendo las notas del medio en el que laboran. Hay poca vinculación entre estas dos esferas de actores. Los portales digitales de noticias parecen estar más dispuestos a interactuar con sus seguidores o a mencionar figuras políticas en sus tuits.

En cuanto a las cuentas identificadas como ciudadanos, algunas pueden recibir el mismo número de interacciones que los medios de noticias. Este es un rasgo distintivo de la comunicación en medios sociales. Anteriormente, el acceso a los medios de comunicación estaba más restringido. Es notable que las cuentas de este tipo de actor hacen uso de todas las formas de interacción disponibles. En este sentido, la reconfiguración de la Comunicación Política está ocurriendo desde la ciudadanía. No obstante, es necesario un mayor involucramiento de los otros dos actores, pues, según lo observado en estas redes, los políticos y los medios repiten las estrategias que emplean en el mundo *offline*. Los actores políticos están en la plataforma para mostrar su presencia y ganar adeptos, mientras que los medios de noticias lo usan para difundir sus notas.

Los nodos con mayor grado de entrada –que suelen pertenecer a figuras políticas relevantes o a medios consolidados– rara vez establecen interacción con aquellos que los mencionan o que responden alguno de sus tuits. Por el contrario, políticos de menor jerarquía y periodistas muestran mayor disposición a intercambiar mensajes.

En este artículo se presenta una identificación de los actores que permite dar seguimiento al tipo de interacciones que tienen entre ellos. Sin embargo, la clasificación de actores puede ser un trabajo complejo, inicialmente manual, en el que el conocimiento previo del experto sobre los usuarios juega un papel fundamental. La automatización del proceso del clasificación no es siempre exacto debido a la limitada información en los perfiles de usuario. Aún así, el trabajo presentado es un acercamiento que puede ser complementado para la clasificación de los usuarios previo a un análisis manual.

5. Conclusiones y trabajo futuro

En este artículo se presentó una propuesta para la recuperación y procesamiento de tuits con el objetivo de visualizar estructuras de interacción entre medios de comunicación, ciudadanos y políticos. Si bien, el trabajo es muy importante en el área hay cuestiones que, por la gran cantidad de datos obtenidos, son difíciles de automatizar. Por ejemplo, la caracterización o perfilado de actores en contextos diversos. Para ello, es importante contar con un proceso manual previo que permita catalogar a los actores de acuerdo con sus funciones.

La clasificación también puede ser complementada con la consideración de otros parámetros, como el comportamiento de publicación, relación entre usuarios seguidos y suscriptores, detección afinada de *pseudo-bots*, así como la implementación de una función de categorización manual por parte del usuario, que, a su vez, sea incorporada en el entrenamiento del clasificador.

Estas afinaciones permtirían, a su vez, una mejor detección de *bots*, algo que se traduciría en un reducción mayor de la cantidad de entradas y un corpus refinado con respecto a la temática de búsqueda.

También se identificó la necesidad de diferenciar aún más la clasificación de las cuentas. Por ejemplo, los medios de noticias pueden ser subdivididos en medios tradicionales, medios digitales y periodistas, ya que su comportamiento en Twitter suele ser distinto.

Por otra parte, es necesario complementar la red con gráficos estadísticos y listas con los actores principales. Otra característica necesaria es el funcionamiento dinámico de los filtros. En otras palabras, la red, y atributos como el grado de entrada y de salida, deben actualizarse con cada filtro aplicado.

Otra área que puede explorarse es el dibujo de redes dinámicas. La observación del despliegue de la red aportaría más datos para la comprensión del fenómeno.

El estudio de las interacciones entre usuarios de Twitter en términos de Comunicación Política debe continuar para incrementar el conocimiento sobre el impacto de la tecnología en los procesos comunicativos de los integrantes de una sociedad. Según lo observado en esta investigación, el análisis de redes es un buen punto de partida que, sin embargo, debe enriquecerse con otras aproximaciones como el análisis de sentimiento o de contenido.

Agradecimientos Agradecemos el apoyo de la coordinación de la Maestría en Diseño, Información y Comunicación (UAM-C) y de la Red Temática en Tecnologías del Lenguaje del CONACyT, México.

Referencias

1. Bruns, A., Moe, H.: Structural layers of communication on twitter. In: Weller, K., Bruns, A., Burgess, J., Mahrt, M., Puschmann, C. (eds.) Twitter and Society, chap. 2, pp. 15–28. Peter Lang (2014)

2. Chadwick, A.: The Hybrid Media System: Politics and Power. Oxford University Press (2013)
3. Cossu, J.V., Abascal-Mena, R., Molina-Villegas, A., Torres-Moreno, J.M., San-Juan, E.: Bilingual and cross domain politics analysis. Avances en la Ingeniería del Lenguaje y del Conocimiento 85, 9–19 (2014)
4. Dubois, E., Gaffney, D.: The Multiple Facets of Influence: Identifying Political Influentials and Opinion Leaders on Twitter. American Behavioral Scientist 58(10), 1260–1277 (2014)
5. Graham, T., Jackson, D., Broersma, M.: New platform, old habits? candidates' use of twitter during the 2010 british and dutch general election campaigns. New Media & Society 18(5), 765–783 (2016)
6. Huang, H., Cao, Y., Huang, X., Ji, H., Lin, C.Y.: Collective tweet wikification based on semi-supervised graph regularization. ACL 1, 380–390 (2014)
7. Jungherr, A.: The logic of political coverage on twitter: Temporal dynamics and content. Journal of Communication 64(2), 239–259 (2014)
8. Kerren, A., Purchase, H.C., Ward, M.O.: Multivariate Network Visualization. Springer International Publishing, Cham (2014), http://link.springer.com/10.1007/978-3-319-06793-3
9. Kruikemeier, S.: How political candidates use twitter and the impact on votes. Computers in Human Behavior (34), 131–139 (2014)
10. Lahuerta-Otero, E., Cordero-Gutiérrez, R.: Looking for the perfect tweet. the use of data mining techniques to find influencers on twitter. Computers in Human Behavior 64, 575–583 (2016)
11. Makazhanov, A., Rafiei, D., Waqar, M.: Predicting political preference of twitter users. Social Network Analysis and Mining 4(1), 1–15 (2014)
12. McNair, B.: An introduction to political communication. Taylor & Francis (2011)
13. Montejo-Ráez, A., Martínez-Cámara, E., Martín-Valdivia, M.T., Ureña-López, L.A.: Ranked wordnet graph for sentiment polarity classification in twitter. Social Network Analysis and Mining 28(1), 93–107 (2014)
14. Neves, A., Vieira, R., Mourão, F., Rocha, L.: Quantifying complementarity among strategies for influencers' detection on twitter. Procedia Computer Science 51, 2435–2444 (2015)
15. Newman, T.P.: Tracking the release of ipcc ar5 on twitter: Users, comments, and sources following the release of the working group i summary for policymakers. Public Understanding of Science (2016)
16. Pfeffer, J.: Fundamentals of visualizing communication networks. China Communications 10(3), 82–90 (2013), http://ieeexplore.ieee.org/xpls/abs{_}all.jsp?arnumber=6488833
17. Smith, M., Rainie, L., Himelboim, I., Shneiderman, B.: Mapping Twitter Topic Networks: From Polarized Crowds to Community Clusters. The Pew Research Center (February 20), 1–57 (2014), http://www.pewinternet.org/2014/02/20/mapping-twitter-topic-networks-from-polarized-crowds-to-community-_clusters
18. Subbian, K., Aggarwal, C.C., Srivastava, J.: Querying and tracking influencers in social streams. In: Proceedings of the Ninth ACM International Conference on Web Search and Data Mining. pp. 493–502. ACM (2016)
19. Wang, J., Cong, G., Zhao, W.X., Li, X.: Mining user intents in twitter: A semi-supervised approach to inferring intent categories for tweets. AAAI pp. 318–324 (2015)

20. Wybrow, M., Elmqvist, N., Fekete, J.D., von Landesberger, T., van Wijk, J.J., Zimmer, B.: Interaction in the visualization of multivariate networks. In: Kerren, A., Purchase, H.C., Ward, M.O. (eds.) Multivariate Network Visualization, chap. 6, pp. 97–126. Springer International Publishing (2014)

ISSN 1870-4069

Visualización de elementos de cienciometría con grafos

Pedro Bello, Meliza Contreras, Diana A. González

Benemérita Universidad Autónoma de Puebla,
Facultad de Ciencias de la Computación, México

{pbello, mcontreras}@cs.buap.mx,
diana_gonznava@hotmail.com

Resumen. En este documento se presenta una herramienta computacional para el análisis de la cantidad de artículos de investigación generada por un conjunto de investigadores; se utilizan los conceptos básicos de la teoría de grafos para representar la cantidad de artículos generados por un investigador y como métrica principal el número de colaboraciones entre un autor y los coautores, así como el análisis de persistencia y continuidad de colaboración científica.

Palabras clave: Cienciometría, colaboración científica, grafo de colaboración.

Visualizing Scientometrics Items with Graphs

Abstract. In this work is presented a computational approach for the analysis of the number of research articles generated by a group of researchers, the basic concepts of the theory of graphs are used to represent the quantity of the items generated by a researcher and as principal metrics is considered the number of collaborations between an author and co-authors as well as analysis of persistence and continuity of scientific collaboration.

Keywords. Scientometrics, scientific collaboration, collaboration graph.

1. Introducción

Con el crecimiento de Internet en los últimos años se han venido desarrollando diversas áreas de investigación que se relacionan con el acceso a la información y el estudio del conocimiento. De esta forma tenemos áreas de interés como la Cybermetrics [1] que estudia los recursos de información en Internet, la Webometrics [14] que estudia los aspectos cuantitativos de construcción y uso de recursos de tecnologías en la web, la Informetrics [2] que maneja aspectos de recuperación de palabras, documentos y bases de datos, Bibliometrics [1] que se encarga del estudio cuantitativo de las publi-

caciones físicas y Scientometrics [3] que se encarga del estudio de la producción científica a través de métodos matemáticos y estadísticos. En la Figura 1 se muestra la relación de las diversas áreas que estudian el acceso a la información.

La Cienciometría (Scientometrics) como ciencia estudia los aspectos cuantitativos de la producción académica; surgió en Europa en 1977 con el nacimiento de la revista Scientometrics. Entre los principales temas que estudia la Cienciometría se encuentran: el crecimiento cuantitativo de la ciencia en base a la producción académica de los investigadores, el desarrollo de las áreas y subáreas, así como la productividad y creatividad de los investigadores. En este contexto se plantea una herramienta computacional que muestre de forma gráfica la producción académica de un grupo de investigadores, utilizando conceptos básicos de grafos. En primera instancia se obtiene un conjunto de datos de prueba de DBLP (Digital Bibliography & Library Project), el cual es un sitio web que posee un enorme repositorio bibliográfico de artículos relacionados con Ciencias de la Computación. El sitio está alojado en la Universidad de Trier, Alemania. Ha evolucionado desde un pequeño servidor web experimental a un popular servicio de datos abiertos para la comunidad en Ciencias de la Computación [15]. La información obtenida viene dada en formato XML (eXtensible Markup Language), que es un lenguaje estándar de marcas que posee una recomendación del World Wide Web Consortium (W3C) y que fue diseñado para almacenar y transportar datos y para ser autodescriptivo. De esta forma se procesan los archivos XML para determinar la producción académica de cada investigador.

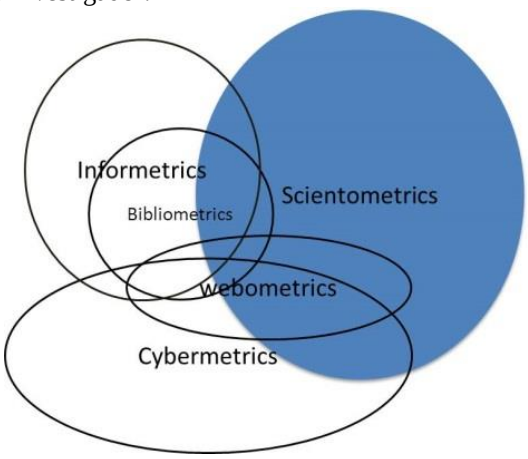

Fig. 1. Relación de las áreas de estudio de la información.

Se realizó una revisión del panorama general de los trabajos relacionados con el tema de estimar el trabajo de los investigadores en la producción de artículos de investigación científica. Por ejemplo, en [4] se muestra un método para el cálculo de citas por autor utilizando matrices, mientras que en [5] se realiza un estudio de la producción académica considerando varios parámetros como la relación entre coautores, co-referencias y co-citas. En [6] se realiza un trabajo para verificar si es válido el método de co-citas para medir el desempeño de los investigadores, debido a que las bases de datos de artículos muchas veces arrojan datos por autores con los mismos apellidos. En [7] se

realiza una comparación entre Web of Science (WoS) de Thomson Reuters y Scopus de Elsevier, y ambos indican que hay que tener precaución debido a que citan problemas con diferentes campos, instituciones, países y lenguajes. En [8] se presenta un estudio que tiene como objetivo examinar la asociación entre el autor y el acoplamiento bibliográfico en 18 áreas temáticas, concluyendo que no hay diferencias significativas en las diversas áreas analizadas. En [9] se realiza un estudio del desarrollo de la producción científica con los datos obtenidos de diversas bases de información en México. Finalmente, en [10] se presenta un modelo para identificar perfiles de usuarios utilizando un grafo de co-ocurrencia.

2. Scientometrics en México

En esta sección mostramos el desarrollo de la ciencia utilizando Scopus que es una base de datos de producción científica a nivel internacional que almacena principalmente: artículos científicos, libros y reportes de conferencias, ofreciendo una visión global del mundo de los resultados de la investigación en los campos de la ciencia, tecnología, medicina, ciencias sociales, artes y humanidades. Scopus ofrece herramientas inteligentes para rastrear, analizar y visualizar la investigación [16].

En la Figura 2 se muestra la producción científica por año, según los resultados se cuenta con 266,378 documentos (consulta realizada el 15 de julio 2016), de los cuales 20,386 corresponden a Ciencias de la Computación, 934 a la Benemérita Universidad Autónoma de Puebla (BUAP). En la intersección de estos conjuntos se tienen 146 artículos que corresponden a los desarrollados en Ciencias de la Computación en la BUAP.

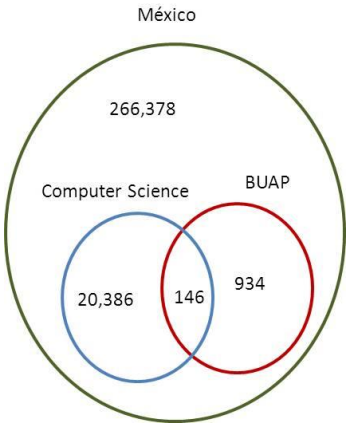

Fig. 2. Producción académica por año en México.

En la Figura 3 se indica que, de acuerdo a Scopus, la BUAP está situada en el ranking 6 de las universidades más productivas del país.

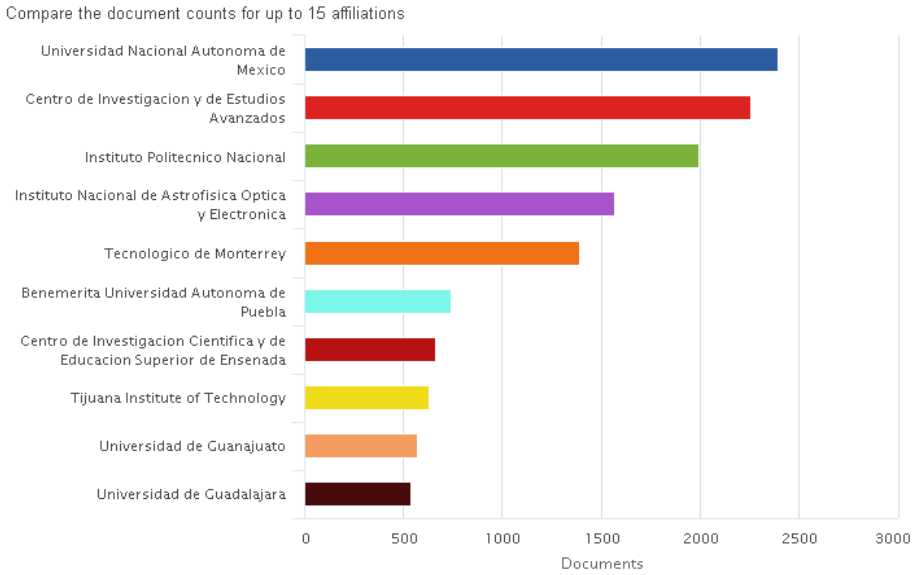

Fig. 3. Producción académica BUAP.

En la Figura 4 se muestra la producción de los documentos reportados por área de conocimiento en México. Se observa que el 24% corresponde a producción de artículos en medicina y solo el 7.7% corresponde al área de Ciencias de la Computación.

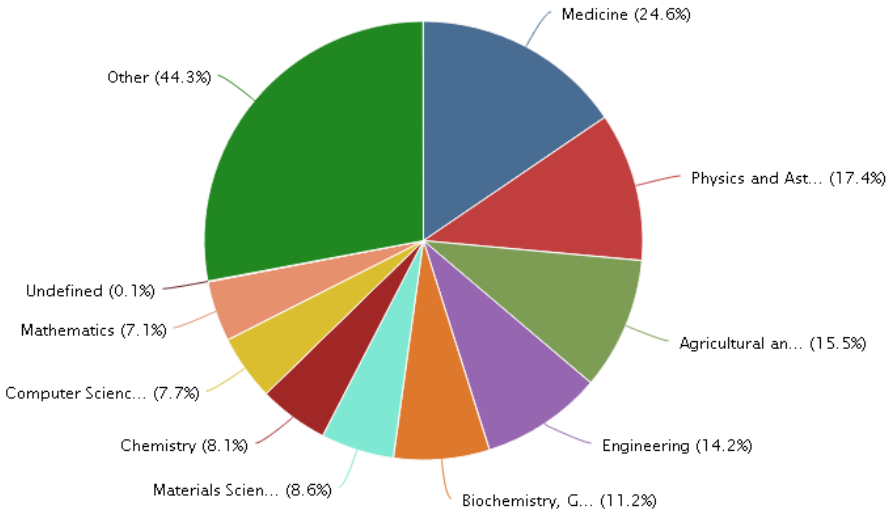

Fig. 4. Producción científica por área.

En la Figura 5 se muestra la producción científica por autor en México, en el área de Ciencias de la Computación.

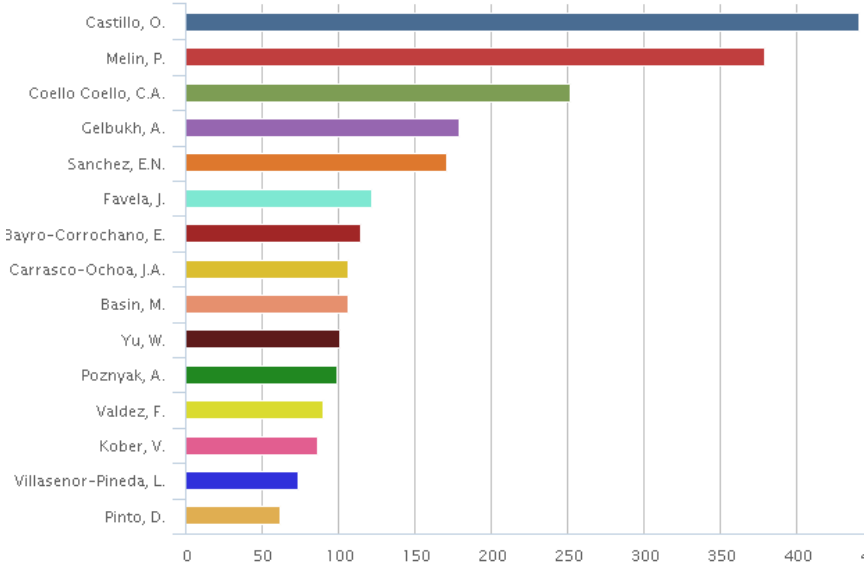

Fig. 5. Producción científica por autor en Ciencias de la Computación (México).

Los sistemas como Scopus permiten realizar un análisis de la producción de documentos científicos. Sin embargo, es posible obtener más conocimiento si, por ejemplo, se desea determinar con que personas se relaciona un investigador determinado. Es justo aquí el punto de estudio del trabajo presentado en este artículo, al proponer una herramienta computacional que aplique diversas métricas para el análisis complementario de la producción académica en México.

3. Metodología

La propuesta para la visualización de elementos de Cienciometría mediante grafos se desarrolló en base a la siguiente metodología de trabajo:

1. Extracción de la información en una base de información de la publicación de artículos.
2. Revisión de la información y el formato XML utilizado.
3. Desarrollo de un programa en PHP que utiliza la librería Vis.js para la representación visual del grafo.
4. Visualización del grafo de colaboración entre autores.

En la Figura 6 se presenta el modelo de la propuesta de la herramienta computacional para determinar la colaboración entre autores en el desarrollo de artículos de investigación. En primera instancia se toman los datos de una base de datos de información de artículos; en este caso se utilizó DBLP que genera archivos en el formato XML, poste-

riormente se creó una aplicación en PHP [17] utilizando una librería grafica denominada Vis.js [18] para la representación del grafo de colaboración, el cual es visualizado a través de un navegador web.

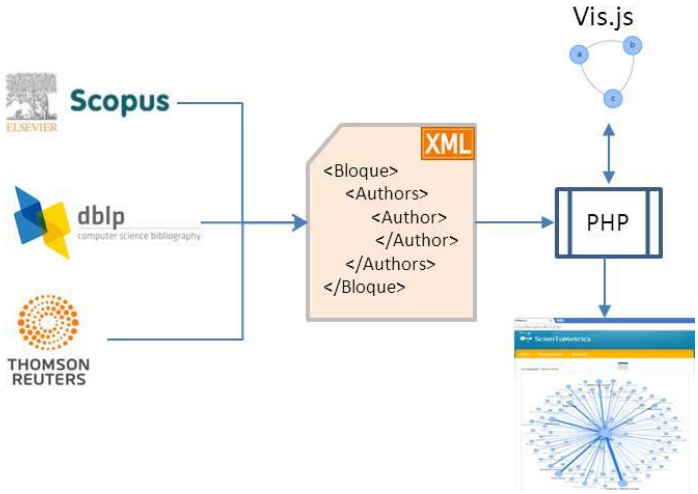

Fig. 6. Modelo de la herramienta computacional propuesta.

3.1. Características cualitativas y cuantitativas

Para medir el desempeño de los investigadores en la producción científica existen diferentes opiniones, por ejemplo [11] propone diferenciar entre indicadores cuantitativos y cualitativos. Las características cualitativas [12] son difíciles de medir mientras que las características cuantitativas utilizan métodos estadísticos. En este trabajo se utilizan medidas cuantitativas básicas.

- Cantidad de publicaciones en revistas científicas y memorias en extenso.
- Cantidad de publicaciones respecto a los coautores.

4. Herramienta computacional propuesta

La propuesta de esta herramienta computacional para el análisis y representación de la colaboración entre autores de artículos de investigación, se basa en la creación de un grafo que acumula la cantidad de artículos producidos por un determinado investigador.

Sea G = (V, E) un grafo sin dirección con un conjunto V de vértices (o nodos) y un conjunto E de aristas (o arcos). El grafo generado tiene las siguientes características:

- Expansión: el nodo central del autor principal acumula la cantidad de artículos en los que ha participado como autor o coautor:

$$v_i = \sum_{n=1}^{\infty} \text{i}, \quad donde\ i\ es\ autor\ o\ coautor. \tag{1}$$

- Amplitud: el arco de cada conexión con los coautores representa la cantidad de artículos en los que ambos han participado. Sea e_{ik} la existencia de una publicación del autor i con el coautor k:

$$Amplitud = w(e_{ik}), donde\ w\ es\ el\ peso\ de\ la\ arista\ (e_{ik}). \tag{2}$$

- Tasa de participación: para grafos relacionados con la colaboración entre los investigadores se calcula el promedio de las participaciones en los artículos:

$$t_{ik} = \frac{v_i}{w_{ik}}. \tag{3}$$

- Relación: en el grafo de colaboración se muestra la relación entre los n autores, donde se diferencian mediante los nodos expandidos, los investigadores que más participan en colaboración con los demás autores:

$$r_i = \sum_{k=1}^{n}(w_{ik}), para\ i = 1..n. \tag{4}$$

Fig. 7. Grafo de colaboración del autor y coautores.

En la Figura 7 se muestra el grafo generado con las publicaciones de la Dra. Mireya Tovar. Como se puede notar, los nodos tienen diferentes tamaños debido a la operación de expansión (1). Además, los arcos de conexión tienen diferente amplitud (2) debido a que existe mayor colaboración entre el autor principal y sus coautores, por ejemplo, la cantidad de artículos entre Mireya Tovar y David Pinto es de 24 como se indica en

la arista correspondiente. En el sistema desarrollado, al acercarse al arco correspondiente se muestra la cantidad de artículos de colaboración. En la Figura 8 se indica el grafo generado de relación entre un conjunto de investigadores; sobresalen los investigadores con mayor productividad (3) y los arcos correspondientes son más amplios ya que representan la mayor colaboración entre pares de investigadores.

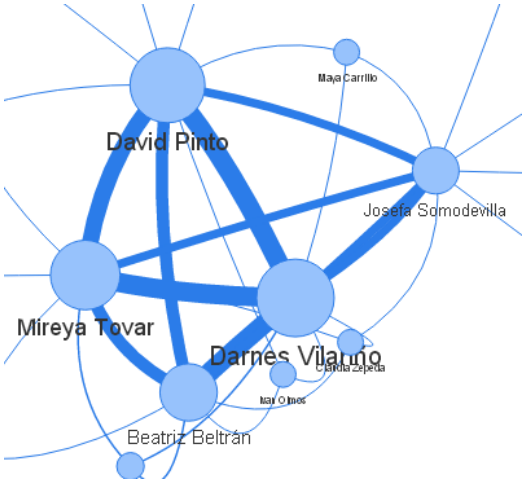

Fig. 8. Grafo que representa la relación entre un grupo de investigadores.

Los grafos de las Figuras 7 y 8 son generados con la herramienta desarrollada. La interfaz principal se muestra en la Figura 9, las opciones que se tiene son: generar el grafo por investigador o por grupo.

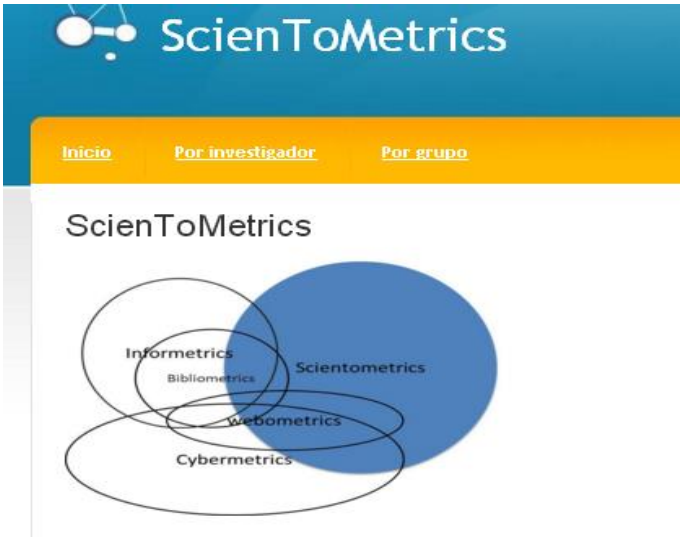

Fig. 9. Vista principal del sistema desarrollado.

En la Figura 10 se muestra el grafo generado de la relación de trabajo entre autores de artículos (4); en el sistema se cuenta con un icono en el lado superior derecho que da como resultado los valores mostrados en la Tabla 1, los cuales corresponden a la cantidad de artículos en colaboración del autor seleccionado.

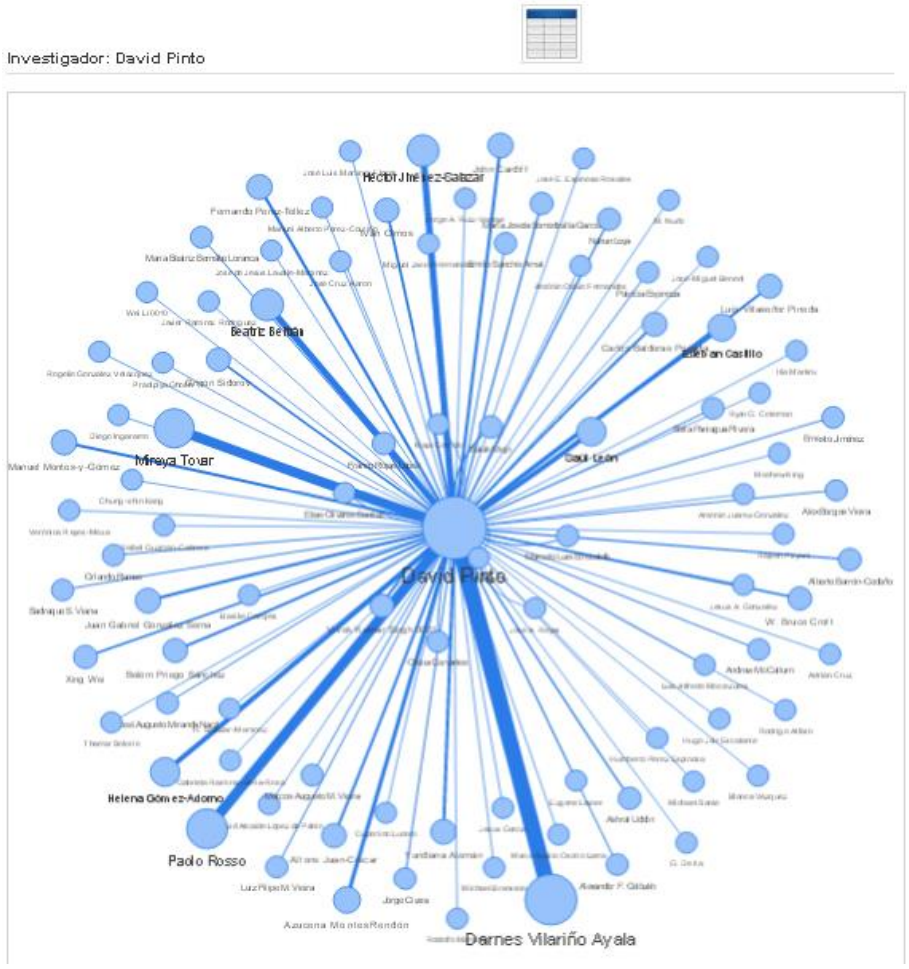

Fig. 10. Grafo de colaboración del Dr. David Pinto a través del sistema Scientometrics.

Tabla 1. Matriz generada para el grafo de relaciones entre investigadores.

	Pinto	Tovar	Beltran	Vilariño	Castro	Somodevilla
Pinto	0	24	15	39	0	3
Tovar	24	0	19	29	1	3
Beltran	15	19	0	18	1	2
Vilariño	39	29	18	0	1	5
Somodevilla	3	3	2	5	0	0

4.1. Estructura de la aplicación

La aplicación propuesta utiliza los grafos como estructura de datos y los archivos xml donde se encuentra almacenada la información de las publicaciones de los investigadores. Se utiliza además PHP como lenguaje de programación y un visor web para la presentación de los resultados. PHP es un lenguaje del lado del servidor, por lo que requiere cualquier servidor web para su ejecución. La principal ventaja de utilizar esta estructura del sistema es que permite ofrecer el servicio a través del servidor a muchas personas con conexión a Internet. Además, facilita la comunicación y extracción de la información para alimentar el sistema. PHP ofrece un conjunto de tecnologías adicionales que permiten la presentación de resultados de forma más atractiva, e interactuar con otros lenguajes de programación y con gestores de bases de datos.

4.2. Pruebas de la aplicación

El sistema desarrollado permite generar el grafo de colaboración de los investigadores. Las pruebas generadas corresponden a leer un archivo xml de DBLP, y se procesa en un programa PHP. Los datos extraídos se muestran en la Tabla 1, donde se indica los datos de un grupo representativo de investigadores de Ciencias de la Computación y se verificó que los resultados obtenidos corresponden a la cantidad de artículos reportados en la base de información correspondiente.

Dada la gran cantidad de información que se utiliza en este tipo de sistemas, es conveniente aplicar una prueba de rendimiento (performance testing) [13] para determinar qué tan rápido responde el sistema con archivos muy extensos y con grupos de investigadores muy amplio.

5. Conclusiones

Se desarrolló una propuesta inicial de un sistema Scientometrics a través de la recopilación de información de una base de información de documentos de investigación en Ciencias de la Computación como DBLP. Dicha propuesta está acompañada de un prototipo con el fin de mostrar que el m4odelo es viable. En este trabajo también se expone que existen sistemas donde generar de forma automática algunos estadísticos que permiten analizar la situación actual de la investigación en México. Sin embargo, es necesario aplicar otro tipo de estrategias de medición, por ejemplo, mostrar la relación de trabajos que existe entre investigadores, indicar cuantos productos se generan en la investigación, indicar la importancia que tiene un investigador respecto a otros, entre otros factores.

El prototipo mostrado sin duda puede ser mejorado, agregando estadísticos que puedan medir de forma cualitativa la investigación y más estadísticos para medir de forma cuantitativa la cantidad de artículos desarrollados por año. Otro de los factores que se pueden medir y determinar gráficamente en este prototipo es conocer el o los colaboradores de un grupo de investigación que participan poco y así integrarlos al trabajo en colaboración con los demás investigadores.

Referencias

1. Thelwall, M., Tsou, A., Weingart, S., Holmberg, K., Haustein, S.: Tweeting links to academic articles. Cybermetrics: International Journal of Scientometrics Informetrics and Bibliometrics, 17 (1), 1–8 (2013)
2. Bar-Ilan, J.: Citations to the Introduction to informetrics. Scientometrics, 82(3), 495–506 (2010)
3. Amara, N., Landry, R: Counting citations in the field of business and management: Why use Google Scholar rather than the Web of Science. Scientometrics, 93 (3), 613–625 (2012)
4. Pinski, G.: Citation based measures of research interactivity. Scientometrics, 2 (4), 257–263 (1980)
5. Krauze, T.K, Mcginnis, R.: A matrix analysis of scientific specialties and Careers in science. Scientometrics, 1 (5-6), 419–444 (1979)
6. Garfield, E.: Is citation analysis a legitimate evaluation tool? Scientometrics, 1 (4), 359–375 (1979)
7. Mongeon, P: The journal coverage of Web of Science and Scopus: a comparative analysis. Scientometrics, 106 (1), 213–228 (2016)
8. Gazni, A., Didegah, F: The relationship between authors' bibliographic coupling and citation exchange: analyzing disciplinary differences. Scientometrics, 107 (1), 609–626 (2016)
9. Ashraf, A., Singh, V. K., Pinto, D., Olmos, I.: Scientometric mapping of computer science research in Mexico. Scientometrics, 105 (1), 97–114 (2015)
10. Espinoza, P., Vilariño, D.; Pinto, D., Somodevilla, J., Tovar M.: Metodología basada en grafos para la identificación de perfiles de usuario. Research in Computing Science, 97, 127–139 (2015)
11. Fehnert, B., Kosagowsky, A., Measuring User Experience - Complementing Qualitative and Quantitative Assessment. In: MobileHCI '08: Proceedings of the 10th international conference on Human computer interaction with mobile devices and services, ACM, 383–386 (2008)
12. Ayala, S, Cuaya, Medina, M, Muñoz, A.: Representación con Restricciones de Medidas Cualitativas: Aplicación a un Problema de Scheduling. LANMR (2006)
13. Somerville, I: Software engineering. Addison Wesley (2011)
14. Webometrics. `https://en.wikipedia.org/wiki/Webometrics` (2016), Accedido el 26 de Abril de 2016
15. DBLP. `http://dblp.uni-trier.de/faq/What+is+dblp.html` (2016), Accedido el 26 de Abril de 2016
16. SCOPUS. `https://www.elsevier.com/solutions/scopus` (2016), Accedido el 6 de Abril de 2016
17. Php. `https://secure.php.net/` (2016). Accedido el 26 de Abril de 2016
18. Visjs. `http://visjs.org/` (2016). Accedido el 26 de Abril de 2016

ISSN 1870-4069

Simplificación de interacciones y la detección de comunidades en una red social

Erick López-Ornelas, Rocío Abascal-Mena

Universidad Autónoma Metropolitana - Cuajimalpa,
Departamento de Tecnologías de la Información, México

{elopez, mabascal}@correo.cua.uam.mx
http://www.cua.uam.mx/

Resumen. Este artículo plantea la utilización del método de *"Mapas Jerárquicos"* para la simplificación de nodos a partir de un conjunto de interacciones en una red social. El corpus utilizado en este artículo está basado en la interacción realizada en Twitter por un conjunto de actores en la destitución del Secretario Mexicano de Hacienda y Crédito Público, el Sr. Luis Videgaray. Al aplicar los mapas jerárquicos es posible simplificar la red social y de esta forma identificar comunidades y actores importantes en el suceso. Un análisis basado en la visualización de las comunidades se realizó para corroborar la pertinencia de la detección.

Palabras clave: Twitter, simplificación de redes sociales, detección de comunidades, visualización de información.

Simplifying Interactions and Detection of Communities in a Social Network

Abstract. In this paper we present the use of the method of *"Hierarchical Maps"* for node simplification using a group of interactions in a social network. The corpus used in this paper is based on Twitter interaction by a set of actors in the destitution of the Mexican Secretariat of Finance and Public Credit, Luis Videgaray. By applying the *"Hierarchical Maps"*, it is possible to simplify the social network and thus identify communities and the major actors in the event. An analysis based on community visualization was conducted to confirm the relevance of the detection.

Keywords. Twitter, social network simplification, community detection, information visualization.

1. Introducción

El uso de plataformas de redes sociales en contextos políticos alrededor del mundo, no solo ha cobrado protagonismo para difundir movimientos sociales, sino que se ha convertido en una importante modalidad de socialización y difusión de información. En el caso de México, la integración de las Tecnologías de la Información y la Comunicación (TICs) y el acceso a las redes sociales digitales ha permitido que los jóvenes y la ciudadanía en general, tengan un medio alternativo de comunicación y poder, de esta manera, interactuar, organizarse y visibilizar sus causas.

Una de las principales redes sociales que se utiliza es Twitter, cuya característica principal es que los mensajes que se envían tienen una longitud máxima de 140 caracteres, llamados tuits, que pueden ser almacenados y categorizados en temas a partir del uso de etiquetas precedidas por el símbolo # (*Hashtag*), mismo que permiten seguir, buscar y encontrar conversaciones relacionadas con un tema en común. La generación de comunidades internas en la red social es un fenómeno natural que se gesta cuando existen muchos actores que generan información relacionada con algún tema en específico.

La identificación de comunidades dentro de una red social es un tema importante ya que, además de simplificar la gran cantidad de información contenida en la red, permite detectar grupos o sectores importantes que interactúan entre sí.

En este artículo hacemos un estudio sobre la destitución reciente del Secretario de Hacienda y Crédito Público, el Sr. Luis Videgaray. Las interacciones en Twitter se gestaron cuando el Presidente de la República, el Lic. Enrique Peña Nieto, anunció el día 7 de septiembre del 2016, cambios importantes en su gabinete, cuando faltan dos años para que termine su periodo en la Presidencia de la República.

Entre los múltiples elementos de análisis que se generaron en las redes sociales bajo el *Hashtag* #Videgaray, decidimos realizar un análisis de las principales comunidades que se gestaron dentro de la red social y de este modo mostrar de manera concisa y simplificada la gran cantidad de información generada en la red. A continuación, mostramos en la Fig. 1 la red generada por las intervenciones alrededor de evento mencionado y usando el *Hashtag* #Videgaray.

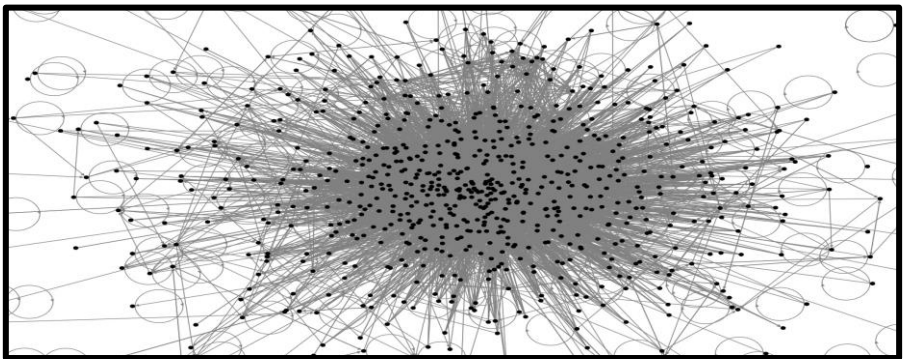

Fig. 1. Red de interacciones #Videgaray el 7 de septiembre del 2016.

La información concentrada en la Fig. 1 corresponde a la extracción realizada el 7 de septiembre del 2016. En la red están representados 850 nodos (tuiteros) y 12666 arcos (interacciones) que fueron realizadas entre ellos. Esta es una red de interacción compleja, por lo que es importante simplificarla para poder realizar un mejor análisis de las interacciones que se gestaron en la red social.

El artículo está organizado de la siguiente manera. En la sección 2 se muestra el objetivo general de la investigación y se muestra el método utilizado para abordar la problemática. En la sección 3 se presenta el estado del arte con algunos de los esfuerzos realizados enfocados en la extracción y análisis de comunidades en una red social. En la sección 4 planteamos y explicamos la técnica utilizada para realizar la detección de comunidades utilizada en este artículo. En la sección 5 se muestra el resultado experimental aplicado y se muestra la simplificación de una red social. Finalmente, en la sección 6 se describen y exponen algunas de las conclusiones y perspectivas de este trabajo.

2. Objetivo y método

El objetivo de este artículo es el de poder analizar y simplificar las interacciones que existen en una red social, a partir de un *trending topic*. Para tal efecto la detección de comunidades es un elemento importante para posteriormente realizar una simplificación de las interacciones.

El método utilizado, tiene una relación directa con la extracción de la información proveniente de Twitter. Se realizó un conjunto de extracciones de tuits generados el día del evento anteriormente mencionado y utilizando el *Hashtag* #Videgaray. Se utilizó la herramienta NodeXL [14] para realizar una primera extracción de los tuits. La visualización mostrada en la Fig. 1 fue realizada utilizando esta herramienta.

Posteriormente, se utilizó el algoritmo basado en los *Mapas Jerárquicos*, el cual se explicará posteriormente, para poder realizar el conjunto de agrupaciones y simplificaciones sobre el grafo inicial correspondiente a la información extraída. Finalmente, una identificación de usuarios relevantes en cada comunidad es propuesta, mostrando la simplificación de las interacciones. El método propuesto es esquematizado en la Fig. 2.

El corpus utilizado está compuesto por 12666 mensajes de Twitter en español escritos por periodistas, políticos y público en general realizados en México el día 7 de septiembre del 2016 entre las 18:00 y 18:05 horas.

Para cada mensaje, la información contenida en el corpus es: su identificador en la red social Twitter, nombre de usuario que lo ha escrito, nombre del usuario a quien hace mención, fecha y hora, el tuit con la descripción textual del mensaje, los hashtags a los que hace referencia en el tuit, el número de seguidores (*Followers*) con que cuenta cada usuario, el número de tuits realizados por cada usuario, la descripción en su perfil de usuario, la ubicación en donde realizó el tuit y la fecha en la que se dio de alta en Twitter.

De toda esta información contenida en el corpus, lo que nos interesa de manera prioritaria son las menciones realizadas, traducidas posteriormente como las interacciones

realizadas por los usuarios, además del número de seguidores y frecuencia de publicaciones de sus tuits.

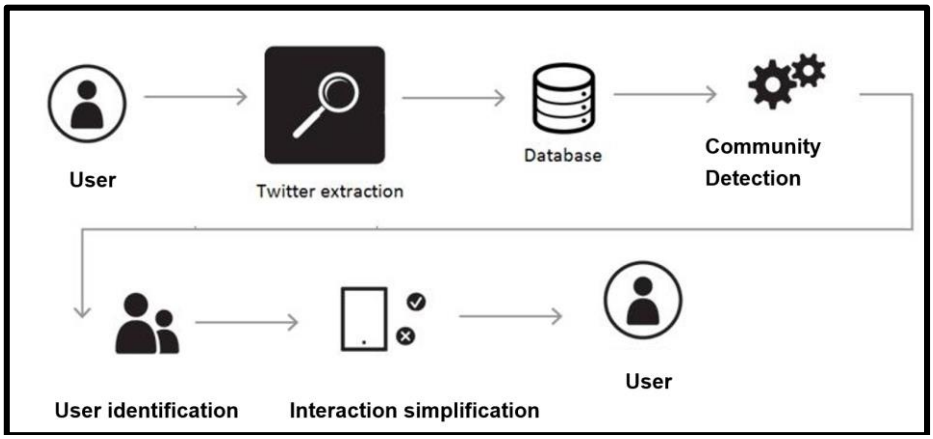

Fig. 2. Método de simplificación de interacciones propuesto.

3. Estado del arte: la detección de comunidades

La estructuración de comunidades, es una propiedad de las redes sociales actuales [5]. Una comunidad puede ser definida como un conjunto de nodos que están más densamente conectados entre ellos que con el resto de la red o que existe una mayor comunicación entre ellos. La importancia de este planteamiento radica en que se espera que los nodos que están contenidos dentro de una misma comunidad compartan atributos, características comunes o relaciones funcionales [4]. Sin embargo, no existe una definición exacta de lo que es, o cómo debería ser particionada la red en una comunidad.

Una partición es la división de una red en comunidades o *clusters*, de modo que todo nodo pertenece a algún *cluster*. Además, estas comunidades pueden estar jerárquicamente estructuradas, es decir, dos o más comunidades al fusionarse pueden formar una comunidad de un nivel superior.

Este tipo de estructuras pueden ser representadas mediante un árbol o dendrograma [4]. Por otro lado, en el caso de que un nodo sea asignado a más de una comunidad hablamos de particiones empalmadas u *overlapping*. Es obvio que conforme crece el número de nodos, dificulta de manera extrema la selección de la mejor partición del grafo.

Cómo encontrar la partición óptima es, probablemente, el problema abierto más importante de la investigación en estructura de comunidades. Una gran variedad de métodos y algoritmos, cada uno de ellos con su propia definición intrínseca de comunidad, han sido desarrollados para intentar extraer la partición óptima de una red. Algunos de ellos tratan de optimizar un índice global de calidad de la partición, como puede ser su *Modularity* [7] o *Surprise* [1]. Otros, sin embargo, utilizan la matriz de adyacencia para

extraer información del grafo, aplicando, por ejemplo, métodos espectrales [13]. Además, estimaciones de máxima verosimilitud [8], o elementos extraídos de la Teoría de la Información [12], son solo unos pocos ejemplos de métodos que han sido aplicados con relativo éxito a la búsqueda de comunidades.

En la literatura existen trabajos interesantes en torno a la detección de comunidades, por ejemplo [2] propuso un algoritmo basado en dos principios: (i) naturaleza intrínseca de las comunidades, y (ii) detección longitudinal, de igual manera en [10], desarrollaron un algoritmo de detección de comunidades solapadas basado en la idea de "amistad" entre los miembros de una comunidad, donde algunos de estos miembros se comportan como líderes de grupo.

En [15] se propone un algoritmo capaz de indicar la influencia de los vértices del grafo contando el número de triángulos que cada vértice comparte con sus vértices adyacentes.

4. Mapas jerárquicos

La definición más general de la comunidad es la de que en la red exista un grupo de nodos que están densamente interconectados. Mientras tanto, desde el punto de vista de la propagación de la información, una comunidad es un grupo de nodos en los que es más probable que se conserve la información en lugar de extenderse.

Teniendo en cuenta que el modelo de propagación fundamental de la información es el "*random walk*" [12], entonces la estructura de una comunidad puede ser identificada mediante una búsqueda local en la estructura. Algunos estudios recientes [3, 6] han demostrado que la modularidad [9], la cual es una función de calidad, es utilizada para encontrar comunidades donde existen grupos de nodos densamente conectados.

En este artículo, utilizamos los "*mapas jerárquicos*" para describir la dinámica de los enlaces y nodos dirigidos, así como de las redes ponderadas para identificar las interacciones locales dentro de la red. Estas interacciones locales permiten calcular el flujo de la información que se puede trasmitir por el nodo, en otras palabras, el grado de interconexión que existe entre dos nodos [12, 11]. En consecuencia, es importante entender el flujo completo de la información en la red.

Un grupo de nodos donde la información fluye de manera rápida y sencilla puede ser agregado y definido como un módulo bien conectado. Los enlaces entre los módulos y las veces que se comunicaron, permiten identificar el grado de conexión entre los módulos, lo que permite realizar una simplificación del grupo generando un módulo y conservando al nodo principal de este módulo o comunidad (Fig. 3).

Esta agrupación radica en la dualidad entre la búsqueda de la estructura comunitaria en las redes y el problema de codificación: encontrar un código eficiente, buscamos entonces, una partición M de nodos n dentro de los módulos m para reducir al mínimo la distancia del llamado "*random walk*". Utilizando el módulo de partición M, la distancia promedio para que la información pase de un nodo a otro está dado por la ecuación 1 [11], la cual está formada por dos términos, el primero es la entropía del movimiento entre los módulos, y el segundo es la entropía del movimiento en los módulos:

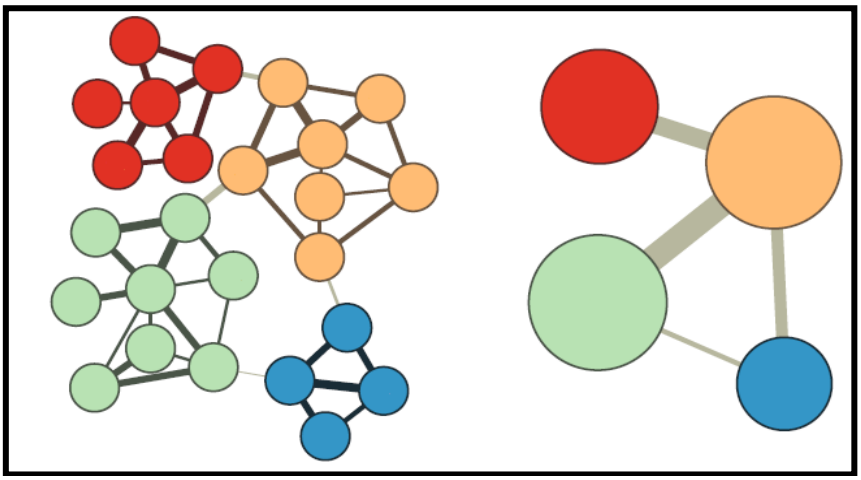

Fig. 3. Esquematización de las agregaciones de nodos en una red.

$$L(\mathbf{M}) = q_{\curvearrowright} H(\mathbb{Q}) + \sum_{i=1}^{m} p^{i}_{\circlearrowright} H(\mathbb{P}^{i}). \tag{1}$$

Los resultados de este método son mostrados utilizando la red generada en el movimiento #Videgaray.

5. Simplificación de la red social

La detección de comunidades, resulta de gran interés para poder analizar las interacciones que se generan en la red. Esta permitirá predecir la intensidad de interacción, en qué etapa se encuentra y cuáles son los actores con mayor peso.

En esta sección aplicamos el algoritmo de *mapas jerárquicos* descrito en la sección anterior para simplificar la red social inicial, lo cual provoca una disminución considerable en el número de nodos visibles.

De los 850 nodos (tuiteros), fueron agrupados y categorizados 30 grandes comunidades. Estas fueron seleccionadas debido a que el porcentaje de la interacción era mayor a 1.0, lo que refleja un conjunto de interacciones importante en la red social. Todas las interacciones menores a 1.0 fueron descartadas.

De estas 30 comunidades identificadas, el actor más representativo fue seleccionado para abanderar la comunidad basado en número de seguidores y número de tuits realizados. De este modo, las 30 comunidades se muestran en la Tabla 1, donde tenemos el módulo de agrupación, el usuario más influyente, la cantidad de información que comparten entre ellos, los que corresponde al porcentaje de las interacciones en la red y el número de nodos que se encuentran agrupados o que son aglutinados dentro de esta gran comunidad.

Tabla 1. Principales comunidades identificadas con el *Hashtag* #Videgaray.

Módulo	Actor representativo	Porcentaje de la interacción	Nodos en la comunidad
1	@epn	13.0	75
2	@cnnmex	16.1	56
3	@cnnee	10.8	66
4	@aristeguionline	7.1	80
5	@adnpolitico	6.0	44
6	@jrisco	1.2	3
7	@pictoline	1.5	2
8	@lvidegaray	5.0	21
9	@jernarovillamil	1.3	4
10	@denisedresserg	4.5	34
11	@lopezdoriga	3.1	12
12	@chumeltorres	1.1	2
13	@nytimes	2.0	85
14	@ap	1.3	2
15	@leonkrauze	2.0	4
16	@washingtonpost	1.2	70
17	@el_pais	1.0	67
18	@josecardenas1	2.3	5
19	@mzavalagc	2.3	14
20	@revistaproceso	2.0	12
21	@jshm00	1.0	1
22	@lydiacachosi	1.3	9
23	@leozuckerman	1.2	8
24	@el_universal_mx	1.2	24
25	@lasillarota	2.0	13
26	@jorgeramosnews	2.6	46
27	@reforma	1.2	79
28	@sncp_mx	3.1	3
29	@josemeadek	3.1	4
30	@sopitas	2.1	5

En este escenario resulta imposible identificar a los usuarios con los que interactúa y mucho menos identificar la importancia en las relaciones. En la Fig. 5 se muestra una simplificación utilizado los *mapas jerárquicos* donde únicamente se muestran las comunidades con las que interactúa. Esta visualización resulta más clara y concisa y permite realizar un mejor análisis de las interacciones existentes en la red.

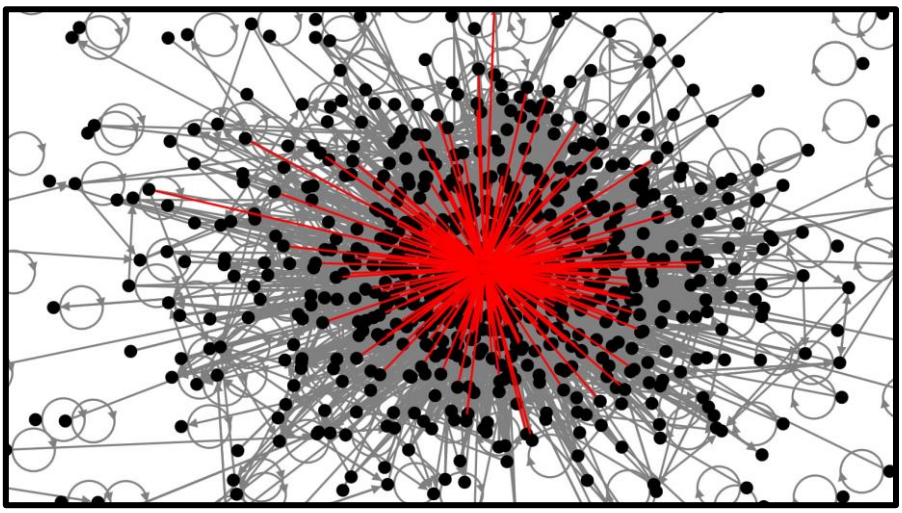

Fig. 4. Enlaces del usuario @cnnmex en la red original.

Al aplicar esta simplificación, inmediatamente podemos apreciar que, en esta interacción, existió una presencia muy importante de los medios de comunicación, incluso podemos determinar quiénes fueron los más activos y con mayor peso (definida por el porcentaje de interacción encontrado) en la red. De este modo es posible identificar los usuarios influyentes en esta interacción.

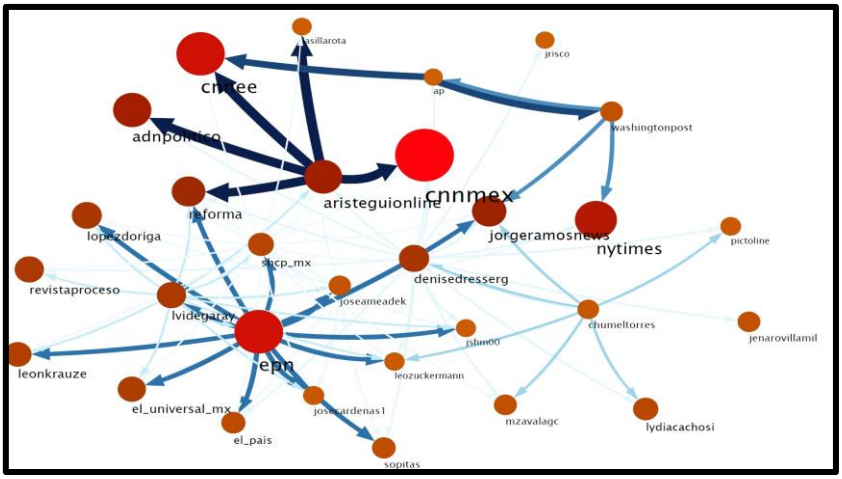

Fig. 5. Simplificación de la red #Videgaray en 30 Módulos.

Otra característica importante a resaltar, es que podemos analizar sub redes de información, esto es, podemos analizar qué es lo que está pasando dentro de la comunidad, cuáles son las interacciones importantes, quiénes son los actores importantes en la sub red y qué fuerzas interactúan dentro de la comunidad. Este análisis es de gran ayuda para determinar la importancia de cada una de las comunidades y sobre todo el poder de influencia no solo de un usuario, sino de la comunidad.

6. Conclusiones

El desarrollo de métodos que puedan detectar estructuras de comunidades en redes sociales, son elementos muy importantes para poder realizar un análisis adecuado, además de que puede desvelar relaciones subyacentes entre los elementos de una red que difícilmente pueden ser vistos en la red original.

En este artículo, hemos realizado una simplificación de una red social generada a partir de un conjunto de interacciones sobre un evento importante (*trending topic*), #Videgaray en nuestro caso, el cual tomó gran importancia a nivel nacional a inicios del mes de septiembre del 2016. La detección de comunidades dentro de la red social permitió identificar un conjunto de características importantes:

- El número de comunidades que se generan internamente.
- La importancia de cada una de las comunidades determinada por el número de interacciones existentes.
- Identificación de actores importantes o influyentes dentro de la comunidad.

En general, el poder analizar las Redes Sociales, resulta un elemento muy poderoso para identificar comportamientos y actores, hacer diagnósticos, descubrir relaciones y también para detectar comunidades. Además, este análisis es susceptible de ser aplicado en áreas diversas como la biología, la política, la computación, la sociología, etc.

La identificación de comunidades presenta un desafío interesante en la precisión con la que se categorizan los múltiples actores o nodos de una red. Sin embargo, los algoritmos actualmente implementados muestran que se está avanzando por el buen camino.

El poder extraer *Hashtags* (#) a partir de una red social como Twitter, permite explorar una gran cantidad de información y determinar tendencias e interacciones existentes.

Como trabajo futuro es interesante el poder comparar la técnica utilizada en este artículo (*mapas jerárquicos*) con algunas otras para poder determinar la confiablidad y la precisión de la misma. Además de poder experimentar esta técnica con otros ejemplos de redes en áreas diversas.

Referencias

1. Aldeco, R., Marín, I.: Deciphering network community structure by surprise. PLoS ONE, 6(9) 8 (2011)

2. Cazabet, R., Amblard, F., Hanachi, C.: Detection of Overlapping Communities in Dynamical Social Networks. Proceedings of the 2010 IEEE International Conference on Social Computing, 309–314 (2010)
3. Delvenne, J. C., Yaliraki, S. N., Barahona, M.: Stability of graph communities across time scales. In: Proceedings National Academy Science USA, 107, 12755 (2010)
4. Fortunato, S.: Community detection in graphs, Physics Reports, 486(3-5), 75–174 (2010)
5. Girvan, M., Newman, E. J.: Community structure in social and biological Networks. In: Proceedings of the National Academy of Sciences of the United States of America, 99(12), 7821–7826 (2002)
6. Kim, Y., Son, S. W., Jeong, H.: Finding communities in directed networks. Phys. Rev. E 81, 016103 (2010)
7. Newman, E. J., Girvan, M.: Finding and evaluating community structure in Networks. Physical Review E - Statistical, Nonlinear and Soft Matter Physics, 69 (2 Pt 2), 16 (2004)
8. Newman, E. J., Leicht, E. A.: Mixture models and exploratory data analysis in networks. In: Proceedings National Academy Science USA, 104(23), 9564–9569 (2007)
9. Newman, E. J.: Modularity and community structure in networks. In: Proceedings National Academy Science, USA 103, 8577 (2006)
10. Palazuelos, C., Zorrilla, M.: FRINGE: A new Approach to the Detection of Overlapping Communities in Graphs. ICCSA, Lecture Notes, 638–653 (2011)
11. Rosvall, M., Axelsson, D., Bergstrom, C. T.: The map equation. Eur Phys J Spec Top, 178 (13), 23 (2009)
12. Rosvall, M., Bergstrom, C.: Maps of random walks on complex networks reveal community structure. In: Proceedings of the National Academy of Sciences of the United States of America, 105(4), 1118–1123 (2008)
13. Shen, H.-W., Cheng, X. Q.: Spectral methods for the detection of network community structure: a comparative analysis. Journal of Statistical Mechanics: Theory and Experiment, 13 (10) (2010)
14. Smith, M. A., Shneiderman, B., Milic-Frayling, N., Mendes Rodrigues, E., Barash, V., Dunne, C., Gleave, E.: Analyzing (social media) networks with NodeXL. In: Proceedings of the fourth international conference on Communities and technologies, 255–264, ACM (2009)
15. Stanoev, A., Smikov, D., Kocarev, L.: Identifying Communities by influence dynamics in social networks. Physical Review (2011)

ISSN 1870-4069

Selección de características para determinar la polaridad de tuits en idioma español a nivel global

Armando Reyes Correa, José Luis Tapia Fabela, Yulia Ledeneva,
René Arnulfo García-Hernández, Rafael Cruz Reyes

Universidad Autónoma del Estado de México,
Toluca, México

armando_reyesc@cbtis162.edu.mx,
joseluis.fabela@gmail.com, yledeneva@yahoo.com,
renearnulfo@hotmail.com, rcruzrey@gmail.com

Resumen. En la actualidad, el internet ha transformado la forma en como las personas se conducen en sus vidas personales y empresariales. Estos cambios son impulsados principalmente por los contenidos generados por los usuarios, mediante opiniones expresadas en forma de texto en las redes sociales. De acuerdo a la Sociedad Española del Lenguaje Natural (SEPLN), el idioma español ocupa el segundo lugar a nivel mundial como lengua materna y como segundo idioma más utilizado en internet, sin embargo, su importancia como lenguaje no corresponde con el mismo nivel de investigación del que es objeto, ya que la mayoría de estas investigaciones se realiza para el idioma inglés. La SEPLN con el fin de promover la investigación para el descubrimiento de nuevos algoritmos y técnicas en análisis de sentimientos en twitter en español, convoca a participar en el taller denominado Análisis de Sentimientos (TASS), que, en su última edición, se invita a participar en dos tareas, Análisis de Sentimientos a Nivel Global y Análisis de Sentimientos a Nivel Aspecto. Este trabajo, se enfoca al problema de la primera tarea. Utilizando los textos que componen el corpus en español otorgado por el TASS 2015, y seleccionando los tres clasificadores más utilizados por el estado del arte Naive Bayes, MSV y J48 mediante el software Weka, en este trabajo se presentan dos modelos, el primero sin ningún tipo de pre-procesamiento y el segundo utilizando características léxicas de los textos, para clasificarlos en seis y cuatro categorías tal como lo define la tarea del concurso.

Palabras clave: Análisis de sentimientos, características léxicas, Naïve Bayes, MSV, J48.

Armando Reyes Correa, José Luis Tapia Fabela, Yulia Ledeneva, et al.

Features Selection to Define the Polarity of Tweets in Spanish at Global Level

Abstract. Nowadays, Internet has transformed the way in how people conduced in their personal lifes and business. This change are mainly proposed by the user contents generated, in opinions expressed by text form in social media. According to Spanish Society of Natural Language Processing (Sociedad Española de Lenguaje Natural, SEPLN), the Spanish language has the second place in importance in the world as mother language and as second language in Internet, however, this fact, does not correspond with the same level of research, due to the most research is for English Language. The aim of SEPLN, through the Workshop of Sentiment Analysis (SA) denominated TASS is to provide a forum where the newest research works in algorithms and in SA techniques in social media, are showed and discussed by scientific and business communities. The last edition of TASS convenes to participate in two tasks, Sentiment Analysis at Global Level, and Sentiment Analysis at Aspect Level. This work is focused in the first task, using the corpus provided by the TASS 2015 workshop, we build two models in Naïve Bayes, Support Vector Machines and J48 classifiers in Weka software, the first one without preprocessing and the second using the main lexical features of the corpus to classify in six and four categories.

Keywords. Sentiment analysis, lexical features, Naïve Bayes, SVM, J48.

1. Introducción

Con el origen de la Web 2.0, Internet contiene grandes cantidades de información generada por el usuario en un ilimitado número de temas. Muchas entidades tales como corporaciones o grupos políticos tratan de obtener conocimiento a través de las opiniones expresadas por los usuarios. Las plataformas sociales tales como Facebook o Twitter han probado ser exitosas para estas tareas, debido al alto volumen de mensajes en tiempo real que se generan y el gran número de usuarios que las utilizan todos los días [1].

Actualmente la gran cantidad de datos almacenados en medios electrónicos, unido al desarrollo tecnológico de las computadoras agrupados bajo el término conocido como “data mining”, tiene como objetivo extraer información y conocimiento útil para aplicarlo en cualquier área productiva apoyando la toma de decisiones [2]. El área de estudio del procesamiento de lenguaje natural encargado de analizar y clasificar textos en polaridades positivas, negativas o neutrales se le denomina análisis de sentimientos, conocido también como minería de opiniones, análisis de subjetividad y orientación de sentimientos [9]. La minería de opiniones [22], en un sentido amplio, se define como el estudio computacional de opiniones, sentimientos y emociones expresadas en el texto.

Detectar sentimientos es considerada una tarea difícil, ya que el problema implica el conocimiento del entorno y contexto donde se ejecuta la opinión, el cual es muy amplio y complejo [23]. Formalmente se dice que una opinión de una característica *c* conlleva un sentimiento asociado, el usuario que emite la opinión es conocido como el emisor de la opinión. De esta forma, una opinión es definida como una quíntupla (o_j, f_{jk}, oo_{ijkl}, h_i, t_i) [8] donde:

o_j - es el objeto de la opinión,

f_{jk} - es una característica del objeto sobre el que se expresa una opinión. Cuando no se detecta ninguna característica, se interpreta como una opinión general, como característica del objeto,

oo_{ijkl} - es la polaridad del sentimiento de la opinión sobre la característica f_{jk} del objeto o_j - positivo, negativo o neutral,

h_i - es el emisor de la opinión,

t_i - es el tiempo en que la opinión es expresada por h_i.

La Sociedad Española del Procesamiento del Lenguaje Natural (SEPLN), una asociación científica, sin fines de lucro que promueve la investigación de todo tipo de actividades relacionadas con el estudio del procesamiento del lenguaje natural en español [7], organiza a partir del año 2012, un taller en Análisis de Sentimientos enfocado en el idioma español sobre textos extraídos de la red social Twitter. Twitter es una plataforma de microblogging donde los usuarios publican mensajes, opiniones y comentarios cuyos contenidos van del rango de los sentimientos personales a publicaciones generales. Las publicaciones en Twitter se conocen como tuits. La característica principal de los tuits es que la longitud máxima del texto es de 140 caracteres [23].

En la última edición de este taller, se organizan dos tareas las cuales son: Análisis de Sentimientos a Nivel Global y Análisis de Sentimientos a Nivel Aspecto. El trabajo presentado, se enfoca en la primera tarea del taller, que consiste en determinar la polaridad global de los mensajes en twitter categorizándolos en seis y cuatro categorías: (P+, P, NEU, N, N+, NONE) y (P, NEU, N, NONE).

Este artículo resume la primera aproximación experimental para determinar la polaridad de tuits en el idioma español, usando el corpus de TASS 2015 y los clasificadores de Weka: J48, Naive Bayes y Máquina de Soporte Vectorial. Las características léxicas extraídas son incluidas como datos de entrada en los clasificadores mencionados, el propósito de este trabajo es analizar las entradas y salidas requeridas por los clasificadores, además de identificar qué ajustes en sus parámetros se logran los mejores resultados.

El resto del artículo se compone de la siguiente manera: Sección 2 resume algunos trabajos publicados en TASS, en la sección 3 se describe la tarea y se analiza el corpus proporcionado, en la sección 4 se explica la parte de experimentación y resultados obtenidos, en la sección 5 se muestra una comparación de los resultados obtenidos con el estado del arte, en la sección 6 se presentan conclusiones y trabajo futuro.

2. Estado del arte

Existen dos enfoques para clasificar comentarios y opiniones en positivos, negativos o neutrales: algoritmos de aprendizaje supervisado y no supervisado. Los algoritmos de aprendizaje supervisado, son utilizados en los problemas donde se conoce a priori el número de clases y miembros representativos de cada clase. La tarea de clasificación de sentimientos, puede ser formulada como un problema de aprendizaje supervisado. Los clasificadores más utilizados en análisis de sentimientos son Naive Bayes (NB), Máquinas de Soporte Vectorial (MSV) y Máxima Entropía. En muchos casos los algoritmos tipo MSV han mostrado una mejora notable sobre Naive Bayes [3].

En el estado del arte estudiado, se ha encontrado que en ediciones pasadas del TASS, el problema de categorizar en seis y cuatro etiquetas se ha abordado mediante el enfoque de aprendizaje supervisado y no supervisado. En el enfoque no supervisado se encuentra el trabajo desarrollado por el equipo SINAI-ESMA en su participación en la edición de TASS 2014 [5], basado en el uso de léxico de opinión y aplicación de una heurística sintáctica, logrando una exactitud en 4 categorías de 0.6456 y en 6 categorías de 0.5360. El equipo del Instituto Politécnico Nacional en el artículo "Análisis de Sentimiento sobre textos en español basado en aproximaciones semánticas con reglas lingüísticas" [6], explica que la clasificación de la polaridad la realiza de acuerdo a un diccionario de orientación semántica, mostrando una exactitud de 0.7648 en 3 categorías y de 0.6261 en 5 categorías.

Analizando los cuatro primeros lugares de las ediciones del TASS 2012 a 2015, se encontró que todos ellos utilizan el enfoque de aprendizaje supervisado. En 2012, el equipo Elhuyar Fundazioa, utiliza un lexicón de polaridad para español a partir de uno en idioma inglés mediante el modelo de unigramas, utilizando como características emoticones negativos y positivos, así como intersecciones positivas y negativas para entrenar un clasificador tipo MSV alcanzando una exactitud de 0.702 en la clasificación de 4 categorías y de 0.641 en seis categorías [12]. En 2013 el equipo DLSI-UA presentó resultados de 0.616, lo que lo ubicó como el primer lugar del concurso, en una clasificación de 6 categorías, utilizando los modelos bigramas y skipgramas apoyándose con un lexicón de sentimientos, para entrenar un clasificador tipo MSV [13]. En ese mismo año, el equipo Elhuyar Fundazioa, segundo lugar de esa edición presentó una metodología creando una lista de vocabulario coloquial a partir de un diccionario de expresiones coloquiales, seleccionando como características los lemas que corresponden a las palabras incluidas en el lexicón de polaridad. El equipo entrena una MSV alcanzado resultados de 0.686 en cuatro categorías y 0.601 en 6 categorías [14]. Para la edición del 2014, el equipo Elhuyar, construye un modelo en Weka, utilizando una MSV, combinando la información extraída de léxicos de polaridad con características lingüísticas. Utiliza un modelo basado en ngramas basados en patrones sintácticos [N+Adj] y [Verb+Noun], con un umbral de 3 ocurrencias para tomar en cuenta los ngramas como características, así mismo, hace uso de los signos de puntuación, tratamiento de la negación y combinación de lexicones. Presenta como resultados 0.6990 para la clasificación en 4 categorías, mientras que para la de seis categorías obtiene un 0.6100 ubicándolo en el segundo lugar [15]. El primer lugar, el equipo EliRF-UPV utiliza el modelo de ngramas de palabras y de lemas, tomando las

características tf-idf de los lemas y de las palabras que aparecen en el tuit, más el número de lemas y palabras que aparecen en el diccionario de lemas y palabra respectivamente. En una tercera aproximación, ejecuta una votación entre 6 sistemas de 1, 2 y 3 gramas de palabras y lemas. Los resultados más altos publicados son 0.6432 para la clasificación de 6 categorías y 0.7089 para la de cuatro categorías. En la última edición del TASS, en el año 2015, el equipo LIF, desarrolla una arquitectura de dos niveles, reproduciendo en el primer nivel 5 sistemas con base en MSV, Redes Neuronales Convolucionales y la implementación del enfoque propuesto por [4]. Este consiste en extraer características doc2vec de los tuits y utilizarlas como entradas en un clasificador del tipo MSV. En el segundo nivel, entrena una MSV, fusionando los resultados del primer nivel para ingresarlos como características. LIF consiguió el primer lugar de esa edición del TASS obteniendo una precisión de 0.6720 para seis categorías y 0.7260 en la clasificación de 4 categorías [9]. El equipo ELiRF-UPV, logró el segundo lugar representando al texto como: votación simple de palabras, lemas y ngramas, además, de utilizar como clasificador una MSV con una precisión de 0.6730 en la clasificación de 6 categorías y 0.7250 en 4 categorías [10].

Otros trabajos analizados es el Sidorov [23], quienes en 2014 desarrollan un lexicón de emociones en español, catalogado como un recurso novedoso para el análisis de emociones en textos, etiquetado con probabilidades que expresan una de las seis emociones básicas. En este trabajo, exploran diferentes configuraciones (tamaño de n-gramas, tamaño del corpus, número de clases de sentimientos, corpus balanceados contra no balanceados, varios dominios) para determinar cómo afectan la precisión de diferentes algoritmos de aprendizaje, utilizando la API de Weka para los clasificadores Naive Bayes, Árboles de Decisión y MSV. Los datos de estudio se componen de tuits en idioma español, recopilado por los autores sobre entidades predefinidas de marcas de teléfonos celulares. Se recopilaron 32,000 tuits y alrededor de 8,000 tuits fueron etiquetados manualmente en una de cuatro categorías, positivo, negativo, neutral o informativo.

En sus resultados concluyen que un conjunto de 3,000 instancias es suficiente como conjunto de entrenamiento para un tema seleccionado; se indica que el unigrama es la mejor longitud utilizada como característica; el número de clases afecta el rendimiento de los clasificadores, reduciendo el número de clases se aumenta la precisión del clasificador; utilizando un corpus no balanceado se logra mejor precisión alcanzando un 0.8580 con un clasificador MSV. Se comprueba que el dominio es parte importante de la precisión alcanzada, ya que, en un experimento, se entrenó los clasificadores utilizando datos del dominio de teléfonos celulares y las pruebas se realizaron con datos del mismo dominio y con un corpus de tuits en el dominio del tema político. Los resultados muestran que el cambio de dominio en la fase de prueba afecta negativamente la precisión pasando de un 0.858 en el mismo dominio en una MSV a un 0.280 en el domino del tema político utilizando el mismo clasificador. En el trabajo desarrollado por Sanzón [21], presentan dos modelos para tratar el corpus otorgado por SemEval 2014, el primero basado en características léxico - sintácticas y el segundo modelo mediante la representación de grafos, selección de características y representación vectorial. Para el primer modelo, se desarrollan dos diccionarios de forma manual sobre emoticones y otro de siglas empleadas en redes sociales,

posteriormente se realiza un pre procesamiento en los datos de entrenamiento y prueba. Finalmente se crean los modelos utilizando los clasificadores Naive Bayes y MSV. Para el segundo modelo, se utiliza el mismo procedimiento de normalización y pre proceso que el primer modelo, los textos se representan mediante grafos de co-ocurrencia no dirigidos utilizando ventanas de tamaño 2 y 3. Los resultados en porcentaje de precisión obtenida en los dos modelos son: En el primer modelo utilizando Naive Bayes 0.4726 y con MSV 0.5658. El modelo de grafos utilizando una MSV, para una ventana de tamaño 2, el mejor resultado obtenido es de 0.4734 y con ventana de tamaño 3 es de 0.4564.

3. Descripción de la tarea y análisis del corpus

Para el TASS 2015, se repite la tarea de ediciones anteriores, que consiste en realizar una clasificación automática para determinar la polaridad global de cada mensaje, validando los modelos sobre el corpus general de prueba. Hay dos evaluaciones diferentes: una basada en 6 etiquetas de polaridad diferente (P+, P, NEU, N, N+, NONE) y otra basada en sólo 4 etiquetas (P, NEU, N, NONE) [7].

El corpus proporcionado por TASS consta de un conjunto de entrenamiento compuesto por 7,219 tuits etiquetados con la polaridad correspondiente a 6 etiquetas. Se muestra la distribución de tuits por su polaridad en el conjunto de entrenamiento

Tabla 1. Distribución de tweets en el corpus de entrenamiento.

Categorías	No. de Tuits	%
+P	1652	22.88%
P	1232	17.07%
NEU	670	9.28%
N	1335	18.49%
+N	847	11.73%
NONE	1483	20.54%

Se cuenta también con un corpus general utilizado como conjunto de prueba compuesto por 60,798 tuits.

La evaluación de los sistemas desarrollados, define el TASS en su sitio web, será utilizando la métrica de Exactitud (Accuracy) [11], la cual evalúa la polaridad correcta asignada a los tuits de acuerdo al estándar de oro. El corpus de prueba, es utilizado para evaluar la exactitud del aprendizaje del modelo, apoyado por la disposición del conjunto de prueba etiquetado. La exactitud de un modelo de clasificación sobre un conjunto de prueba es definida como [16]:

$$Exactitud = \frac{Número\ correcto\ de\ clasificaciones}{Número\ total\ de\ casos\ de\ prueba}.$$

La matriz de confusión generada, será utilizada para evaluar precisión, recuerdo y medida - F1 para cada categoría individual. La metodología utilizada como primer aproximación se representa en la figura 1.

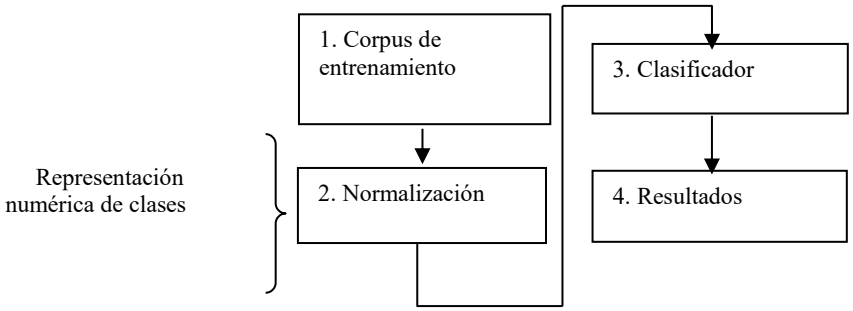

Fig. 1. Metodología propuesta.

4. Experimentación

La primera aproximación de solución consistió en utilizar el corpus de entrenamiento importando a Excel el archivo en formato XML. Se eliminó la información que no se requiere para el análisis de sentimientos, permaneciendo solamente la columna etiquetada con el texto *content*, que es el tuit a analizar y la columna *value*, que contiene las etiquetas de las clases (+P, P, NEU, N, N+ y NONE).

Con este formato se utilizaron los clasificadores J48 y Naive Bayes, posteriormente se cambió el formato de la columna *value* para que las clases utilizaran una representación numérica de la siguiente forma: 0 = NONE, 1 = NEU, 2 = N, 3 = N+, 4 = P y 5 = +P y poder experimentar con el clasificador MSV. Se inicia el programa de análisis de datos Weka versión 3.6 y se importó el archivo .csv que contiene los datos de entrenamiento etiquetados del corpus TASS 2015 otorgándonos una vista de la distribución de clases. Se seleccionan los clasificadores J48, LibSVM y NaiveBayes con el parámetro porcentaje de división al 80% (*percentage split*), que se refiere a que, de los datos ingresados, se tomará el 80% de entrenamiento y el 20% restante como datos de prueba. Los resultados de este experimento se muestran en la tabla 2:

Tabla 2. Resultados de clasificación en seis categorías, sin pre procesamiento, utilizando 80% de división.

Clasificador	J48	LibSVM	Naive Bayes
Resultado	.2458	.2437	.2458

El segundo experimento consistió en utilizar la validación cruzada de 10 folders eligiendo los mismos clasificadores sin pre-procesamiento. Los resultados se muestran en la tabla 3.

Tabla 3. Resultados de clasificación en seis categorías, sin pre procesamiento, utilizando validación cruzada de 10 folders.

Clasificador	J48	LibSVM	Naive Bayes
Resultado	.2317	.2288	.2317

Para la construcción del segundo modelo, se dividió el corpus de entrenamiento en 6 diferentes archivos, cada uno, contiene los textos que corresponden a una clase. Se programaron componentes en Python utilizando expresiones regulares para extraer características estadísticas y léxicas, como la frecuencia por tuit de hashtags, urls, y menciones a usuario; de las características léxicas, se utilizó una lista de emoticones de Wikipedia, categorizados como Sonrisa, Risa y Tristeza [17], con el fin de obtener una frecuencia por tuit. El estado del arte indica que el uso de mayúsculas representa énfasis al tratar de transmitir una idea, opinión o sentimiento, de tal forma que se tomaron en cuenta las frecuencias por cada tuit de las palabras que comienzan en mayúscula, así como palabras que se encuentran escritas en su totalidad en mayúsculas. El archivo de características es utilizado para ingresarlo a Weka, construir el modelo de clasificación con los mismos parámetros que el modelo descrito anteriormente; con el fin de comparar los resultados obtenidos. Cabe hacer mención, que los tuits hasta este momento, no han recibido ningún tipo de pre procesamiento.

Tabla 4. Resultados de clasificación en seis categorías, sin pre procesamiento, utilizando 80% de división con las características léxicas extraídas.

Clasificador	J48	LibSVM	Naive Bayes
Resultado	.2832	.2853	.2610

Utilizando las mismas características, pero evaluando mediante validación cruzada de 10 folders, se obtienen los siguientes resultados:

Tabla 5. Resultados de clasificación en seis categorías, sin pre procesamiento, utilizando validación cruzada de 10 folders con las características léxicas extraídas.

Clasificador	J48	LibSVM	Naive Bayes
Resultado	.2811	.2878	.2577

5. Comparación de resultados

En la figura 2, se muestran los resultados obtenidos en este trabajo y se comparan con los alcanzados por los equipos participantes en el TASS 2015, que obtuvieron los cuatro primeros lugares en el taller y los obtenidos en nuestros experimentos. Los

experimentos son evaluados con la métrica de exactitud (Accuracy), de acuerdo a lo estipulado en la página oficial del concurso TASS 2015[1].

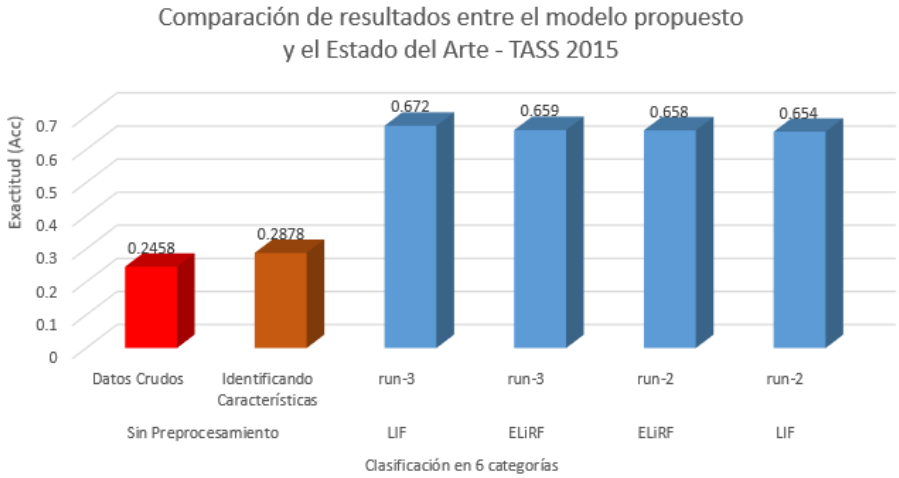

Fig. 2. Comparación de resultados con los del estado del arte.

Como se puede observar, el modelo propuesto aún necesita ser ajustado para alcanzar mejores resultados.

6. Conclusiones y trabajo futuro

El estado del arte indica que las características especiales del lenguaje de Twitter, requieren un tratamiento especial para analizar los textos. La sintaxis particular, menciones a usuarios, urls, hashtags, emoticones, oraciones pobres en gramática, modismos, entre otras, conducen a una caída en el rendimiento de las herramientas tradicionales del PLN [18]. Existe una propuesta en los trabajos [19] y [20] para la normalización del texto que presentan estas características.

Este artículo resume la experimentación realizada con los datos del corpus TASS 2015, utilizando como primera aproximación los clasificadores J48, Naive Bayes y MSV incluidos en Weka. Se observó que, al extraer frecuencias de características, los resultados han mejorado. El estado del arte indica que es importante el pre procesamiento en este tipo de textos, por lo que en primera instancia podemos concluir, que la etapa de pre procesamiento puede ser un factor clave en el tratamiento de la información para alcanzar mejores resultados.

Como trabajo futuro, se propone agregar como características las frecuencias de término y la frecuencia inversa del término (*tf-idf*), representación mediante unigramas y bigramas experimentando con los mismos clasificadores y los mismos parámetros, comparando el rendimiento de los modelos en base a sus resultados. Se considera

[1] http://www.sepln.org/workshops/tass/2015/tass2015.php

importante, que los experimentos futuros incluyan algún tipo de pre procesamiento, por lo tanto se pretende utilizar la herramienta desarrollada por [23], con el fin de apoyar esta etapa buscando más características léxicas para mejorar los resultados hasta ahora alcanzados.

Agradecimientos. Trabajo soportado por CONACYT.

Referencias

1. Cesteros, J. S., Almeida, A., de Ipiña, D. L.: DeustoTech Internet at TASS 2015: Sentiment Analysis and Polarity Classification in Spanish Tweets. In; TASS@ SEPLN, 2, 23–28 (2015)
2. Rodríguez, M., Alvarez, J., Mesa, J., González, A.: Metodologías para la realización de proyectos de Data Mining. Ponencia presentada en el VII Congreso Internacional de Ingeniería de ProyectosK Elissa (2005)
3. Valverde, J., Tejada, J., Cuadros, E.: Comparing Supervised Learning Methods for Classifying Spanish Tweets. CEUR-WS, 87–92 (2015)
4. Le, Quoc V., Mikolov, T.: Distributed representations of sentences and documents. In: ICML, 14, 1188–1196 (2014)
5. Zafra, J., María, S., Martínez Cámara, E., Valdivia, M., Teresa, M., Ureña López, L. A.: SINAI-ESMA: An unsupervised approach for Sentiment Analysis in Twitter. In: Proceedings of the TASS workshop at SEPLN, 16–19 (2014)
6. Hernández Petlachi, R., Xiaoou, L.: Análisis de sentimiento sobre textos en español basado en aproximaciones semánticas con reglas lingüísticas. In: Proceedings of the TASS workshop at SEPLN (2014)
7. Villena-Román, J., Garcıa-Morera, J., Garcıa-Cumbreras, M. A., Martınez-Cámara, E., Martın-Valdivia, M. T., Urena-López, L. A.: Overview of TASS 2015. In: TASS 2015 Workshop on Sentiment Analysis at SEPLN, 1397, 13–21 (2015)
8. Liu, B.: Sentiment Analysis, Mining opinions, sentiments and emotions. Cambridge University Press, First edition (2015)
9. Rouvier, M., Benoit, F.: LIF @ TASS 2015: Deep models are very coooooollllll for sentiment analysis (2015)
10. Hurtado, L. F., Pla, F., Buscaldi, D.: ELiRF-UPV en TASS 2015: Análisis de Sentimientos en Twitter. In: Proceedings of TASS 2015, Workshop on Sentiment Analysis at SEPLN, 35–40 (2015)
11. Sepln.org.: TASS 2016. @ SEPLN, Recuperado de: `http://www.sepln.org/workshops/tass/2016/tass2016.php` Fecha de Acceso 30 Agosto de 2016 (2016)
12. Saralegi, X., San Vicente, I.: Tass: Detecting sentiments in Spanish tweets. In: Workshop on Sentiment Analysis at SEPLN (TASS), SEPLN, 9 (2012)
13. Fernández, J., Gutiérrez, Y., Gómez, J. M., Martínez-Barco, P., Montoyo, A., Munoz, R.: Sentiment analysis of Spanish tweets using a ranking algorithm and skipgrams (2013)
14. Urizar, X. S., Roncal, I. S. V.: Elhuyar at TASS 2013. In: Proceedings of the Workshop on Sentiment Analysis at SEPLN (TASS 2013), 143–150 (2013)
15. San Vicente Roncal, I., Urizar, X. S., Looking for features for supervised tweet polarity classification. In Proceedings of the TASS workshop at SEPLN (2014)
16. Liu, B.: Web data mining: exploring hyperlinks, contents, and usage data. Springer Science & Business Media (2007)
17. Anexo: Emoticonos (s.f.). En Wikipedia, Recuperado el 16 de septiembre de 2016 de https://es.wikipedia.org/wiki/Anexo:Emoticonos
18. San Vicente Roncal, I., Urizar, X. S.: Looking for features for supervised tweet polarity classification. In: Proceedings of the TASS workshop at SEPLN (2014)

19. Brody, S., Diakopoulos, N.: Cooooooooooooooollllllllllll!!!!!!!!!!!!!!: using word lengthening to detect sentiment in microblogs. In Proceedings of the conference on empirical methods in natural language processing, 562–570 (2011)
20. Han, B., Baldwin, T.: Lexical normalization of short text messages, Makn sens a# twitter. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies 1, 368–378 (2011)
21. Sanzón, Y. M., Vilariño, D., Somodevilla, M. J., Zepeda, C., Tovar, M.: Modelos para detectar la polaridad de los mensajes en redes sociales. Research in Computing Science, Avances en la Ingeniería del Lenguaje y del Conocimiento, 29 (2015)
22. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Foundations and trends in information retrieval, 2(1-2), 1–135 (2008)
23. Sidorov, G., Miranda-Jiménez, S., Viveros-Jiménez, F., Gelbukh, A., Castro-Sánchez, N., Velásquez, F., Gordon, J.: Empirical study of machine learning based approach for opinion mining in tweets. In: Mexican International Conference on Artificial Intelligence, 1–14 Springer Berlin Heidelberg (2012)
24. Liu, B.: Sentiment Analysis and Subjectivity. Handbook of natural language processing, 2, 627–666 (2010)

ISSN 1870-4069

Sistema de reconocimiento y clasificación de señas para el lenguaje español

Jorge Cerezo[1], Griselda Saldaña[1], Mario M. Bustillo[2], Apolonio Ata[2], J. Andrés Vázquez[2], Beatriz Bernabé[2], Gerardo Martínez [2]

[1] Universidad Tecnológica de Puebla,
Ingeniería en Tecnologías para la Automatización, México

[2] Benemérita Universidad Autónoma de Puebla,
Facultad de Ciencias de la Computación, México

{jorge.cerezo, griselda.saldana}@utpuebla.edu.mx,
bustillo1956@hotmail.com,
{apolonio.ata, andrexsol, beatriz.bernabe}@gmail.com,
gguzman@cs.buap.mx

Resumen. En este trabajo se presenta un sistema para reconocimiento y clasificación de las letras que componen el lenguaje de señas en español, diseñado para apoyar en el entrenamiento de personas sordomudas y facilitar su comunicación con el resto de las personas. Se basa en un guante de bajo costo que captura el movimiento de la mano, el cual contiene un acelerómetro en cada dedo para detectar posición, y está conectado a una tarjeta de adquisición de datos. La información de los sensores se envía de forma inalámbrica a una computadora que contiene una interface desarrollada en LabVIEW, donde se genera una base de datos de símbolos. Para el reconocimiento de las letras, se aplicó un tratamiento estadístico a los datos en dicha base obteniendo una precisión superior al 96% independientemente del usuario.

Palabras clave: Lenguaje de señas, clasificación automática, guante.

Recognition and Classification of Sign Language for Spanish

Abstract. In this paper it is presented a computational system for recognition and classification of letters for sign language in Spanish, designed for helping deaf-mute people to communicate with other people. A low-cost glove that capture the hand movements has been constructed. This one contains an accelerometer for each finger which allows to detect its position by using an acquisition data card. Sensor information is sent wirelessly to a computer having a software interface, developed in LabVIEW, in which the symbols dataset is generated. For the automatic recognition of letters, we have applied a statistical treatment to the dataset obtaining an accuracy greater than 96%.

Keywords. Signs language, machine learning, glove.

1. Introducción

El reconocimiento de gestos ha recibido atención de muchas áreas de investigación tales como interacción humano-computadora, la realidad virtual, la tele-manipulación y el procesamiento de imágenes. Otra área de aplicación es la interpretación del lenguaje de señas [1]. Entre los tipos gestos, el lenguaje de señas es uno de los más estructurados, usualmente cada gesto está asociado a un significado predefinido. Por otra parte, la aplicación de fuertes reglas de contexto y gramática hace que el lenguaje de señas sea más fácil de reconocer [2].

De acuerdo a la tecnología de sensado empleada para capturar los gestos, existen dos aproximaciones principales para el reconocimiento de señas, una basada en técnicas de visión [3], donde se sigue el movimiento de la mano y se interpreta la seña correspondiente [4, 5] y otra basada en guantes [6] que cuentan con sensores que capturan la rotación y movimiento de la mano y los dedos [7]. El reconocimiento en base a sensores tales como acelerómetros y giroscopios ofrecen las siguientes ventajas: a) dado que los sensores de movimiento no se ven afectados por el entorno, el reconocimiento es más adecuado que el reconocimiento basado en visión en entornos complejos b) están unidos a un usuario, proporcionando así una mayor cobertura, y c) las señales se pueden adquirir de forma inalámbrica [8].

En trabajos previos se han utilizado de manera exitosa guantes como elementos para reconocimiento de señas [10], en [1] se presenta un sistema para reconocimiento de las 23 letras del lenguaje vietnamita, empleando un guante con acelerómetros MEMS, cuyos datos se transforman a ángulos relativos entre los dedos y la palma. Para el reconocimiento de las letras utiliza un sistema de clasificación basado en lógica difusa. En [11] se reporta un guante basado en acelerómetros y sensores mioeléctricos, los cuales permiten detectar de forma automática el punto inicial y final de segmentos significativos de los símbolos mediante la intensidad de los sensores mioeléctricos. Para obtener el resultado final, utiliza árboles de decisión y modelos ocultos de Markov. La funcionalidad del sistema se demuestra al clasificar los 72 símbolos del lenguaje de señas chino.

En este trabajo se presenta la implementación de un sistema entrenador del lenguaje de señas del alfabeto en español para sordomudos. Consta de un dispositivo tipo guante con un acelerómetro conectado a cada dedo. Las salidas de los sensores pasan a una tarjeta de adquisición que envía los datos de forma inalámbrica a una computadora donde reside una interface en LabVIEW.

Los datos recolectados se mantienen en una base de datos de símbolos, donde a diferencia de [7] esta información se clasifica empleando un método estadístico. Una vez que se discriminan los símbolos sin ambigüedad, el sistema puede utilizarse para entrenamiento de personas sordomudas, quienes desde otra interface en LabVIEW, pueden realizar cada una de las letras del alfabeto en español y comprobar si lo hacen de forma correcta.

El resto del documento está organizado de la siguiente manera. En la sección 2 se presenta una descripción del sistema, haciendo énfasis en la implementación del guante y el funcionamiento de los sensores. En la sección 3 se presentan el mecanismo de clasificación de los datos. En la sección 4 se presentan las pruebas realizadas al sistema así como algunos resultados obtenidos para finalmente en la sección 5 presentar las conclusiones y trabajo futuro.

2. Descripción del sistema

El sistema consta de tres elementos, un guante instrumentado con acelerómetros analógicos que puede enviar información de forma inalámbrica, y dos programas en labVIEW, el primero para la captura de muestras y el segundo para el entrenamiento de personas en la realización de símbolos. Los programas cuentan con una interface gráfica que es muy intuitiva y permiten a cualquier usuario interactuar con el sistema**.**

2.1. Construcción del guante

El diseño del guante se basa en la utilización de acelerómetros, en este caso los ADXL335 ya que son de bajo costo y consumen poca potencia. Dichos acelerómetros proporcionan una medida de la posición de los dedos en tres ejes con un formato serial (x, y, z), los acelerómetros del guante proporcionan datos crudos que se envían a la tarjeta de adquisición en un formato de vector y se envían a la computadora central a través de un dispositivo Xbee.

Fig. 1. Estructura del guante.

2.2. Captura de las muestras

La computadora tiene desarrollado un programa en LabVIEW que se utiliza para capturar los datos correspondientes a cada una de las letras del alfabeto en español y almacenarlo en una base de datos. Para ello se recurrió a un grupo de 25 personas sordomudas quienes realizaron un total de 50 veces cada letra. La interface de usuario realizada para la captura de las muestras se observa en la figura 2.

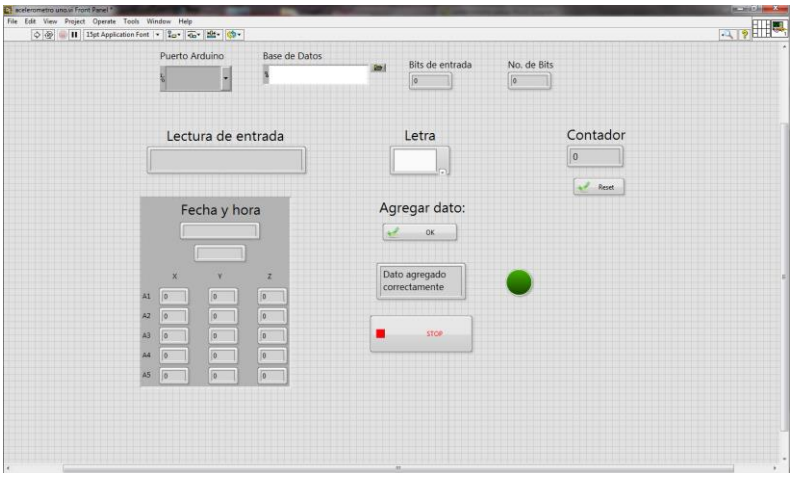

Fig. 2. Interface de usuario para la captura de datos.

Fuera de línea, se utilizan estos datos para realizar un proceso de clasificación donde cada subclase corresponde a una letra en particular. Para la operación en línea el usuario a entrenar accede a otra interface de usuario donde se le indica si es que está realizando de forma adecuada cada letra.

El usuario ejecuta una letra y a continuación se hace una lectura de los datos del guante, la información se compara con la información obtenida del sistema entrenador. Una vez que se reconocen cada una de las lecturas X, Y, Z de cada acelerómetro, se muestra la letra correspondiente en la pantalla, y así el usuario puede corroborar que la realiza de manera correcta y repite el proceso con un nuevo símbolo. Si lo desea, puede proceder a formar una palabra.

3. Clasificación

Una vez capturado los datos, podemos utilizarlos para construir un modelo de clasificación que pueda ser utilizado posteriormente para identificar una seña y asociarla automáticamente a una determinada letra. Las lecturas X, Y y Z que se obtuvieron de cada uno de los cinco acelerómetros se utilizan como características para la construcción del modelo de clasificación. En particular, hemos hecho experimentos con los siguientes tres clasificadores:

a. J48: Es un clasificador del tipo de árbol de decisión: El algoritmo J48 es una implementación del algoritmo C4.5, uno de los algoritmos de minería de datos que más se ha utilizado en multitud de aplicaciones.
b. SMO: El nombre proviene del inglés “Sequential minimal optimization”, y es un algoritmo para solucionar el problema de programación cuadrática que surge durante el entrenamiento de las máquinas de soporte vectorial. Fue inventado en 1998 por John Platt [12] y es ampliamente usado en la actualidad.
c. El perceptrón multicapa es una red neuronal artificial (RNA) formada por múltiples capas, esto le permite resolver problemas que no son linealmente separables, lo cual es la principal limitación del perceptrón (también llamado perceptrón simple).Los resultados obtenidos en los experimentos se muestran en la siguiente sección.

4. Pruebas y resultados

En esta sección se describen características asociadas al corpus de entrenamiento, así como la metodología de evaluación y los resultados obtenidos.

4.1. Conjunto de datos

En la Tabla 1, se puede observar el número de muestras que han sido capturadas para cada uno de los símbolos considerados en el corpus de entrenamiento. El número mínimo de muestras es de 47 para la letra ‘m’, mientras que el máximo número de muestras fue de 96 para la letra ‘f’. La media de muestras es de 55.12. En total se capturaron 1,378 muestras.

Tabla 1. Cantidad de muestras por cada símbolo del alfabeto.

Letra	Muestras	Letra	Muestras	Letra	Muestras
a	58	i	51	q	50
b	51	j	48	r	65
c	57	k	48	s	66
d	51	l	49	t	56
e	50	m	47	u	53
f	96	n	59	v	51
g	51	o	56	w	51
h	48	p	66	x	50
				y	50

4.2. Metodología de evaluación

El proceso de evaluación considera el uso del corpus de entrenamiento para validar la exactitud en la identificación de las letras, usando los tres modelos de clasificación automática planteados con anterioridad.

Se divide cada conjunto de muestras de cada letra en 10 particiones y se ejecutan diez iteraciones usando un 90% de los datos para entrenamiento y el 10% restante para pruebas en un proceso denominado 10-fold cross-validation y leave-one out.

Los resultados obtenidos en los tres clasificadores, así como la discusión de dichos resultados se presentan en la siguiente sección.

4.3. Resultados obtenidos

En la Tabla 2 se muestran los resultados obtenidos para cada uno de los clasificadores. Como puede observarse, es el clasificador basado en perceptrón multicapa el que obtiene los mejores resultados, con una exactitud superior al 97%. De las 1,378 muestras clasificadas, solamente clasificó incorrectamente un total de 36 instancias, dando como resultado un error del 2.61%. De hecho, supera en 5 puntos porcentuales al clasificador SMO y en 8 puntos porcentuales al clasificador J48.

Estos resultados muestran que el grado de exactitud es elevado y suficiente para el proceso de clasificación de letras basado en lenguaje de señas.

Tabla 2. Comparativa de resultados obtenidos entre los tres clasificadores.

	J48		**SMO**		**Perceptrón multicapa**	
Instancias clasificadas	Cantidad	Porcentaje	Cantidad	Porcentaje	Cantidad	Porcentaje
Correctamente	1,227	89.04%	1,276	92.60%	1,342	97.39%
Incorrectamente	151	10.96%	102	7.40%	36	2.61%

Es necesario, sin embargo, realizar un análisis de los tiempos de ejecución necesarios por cada algoritmo para construir el modelo de clasificación, a fin de verificar su pertinencia de uso en sistemas de tiempo real. En la Tabla 3 se muestran dichos resultados.

Tabla 3. Comparativa de tiempos de construcción del modelo de clasificación.

	J48	**SMO**	**Perceptrón multicapa**
Tiempo (segundos)	0.19	2.45	17.86

Como puede observarse, el nivel de exactitud es inversamente proporcional a los tiempos de ejecución necesarios para construir el modelo. En realidad, los casi 18 segundos necesarios por el clasificador basado en perceptrón multicapa no resulta ser prohibitivo para construir un modelo de clasificación. De hecho, los tiempos de

evaluación de las instancias de prueba son en milésimas de segundos para cualquiera de los tres clasificadores probados.

5. Conclusiones

En este trabajo se presentó un guante basado en acelerómetros que permite el entrenamiento de personas sordomudas para escribir las letras del alfabeto en español. El tratamiento de los datos se realizó de manera estadística, permitiendo precisión en el proceso de clasificación y hace al sistema ser independiente del usuario y permite la detección de símbolos aun si éstos no tienen una forma perfecta.

Los experimentos realizados con tres métodos de clasificación automática muestran que la precisión obtenida en la identificación de los símbolos es mayor al 89%. En particular, el algoritmo basado en redes neuronales utilizando un perceptrón multicapa obtuvo el mejor resultado, con una exactitud del 97%. Como trabajo futuro se pretende incluir un sintetizador de voz para producir palabras tras detectar un conjunto válido de símbolos del alfabeto.

Agradecimientos. Los autores agradecen al PRODEP por el apoyo para la realización de este trabajo.

Referencias

1. Bui, T. D., Nguyen, L. T.: Recognizing Postures in Vietnamese Sign Language with MEMS Accelerometers. IEEE Sensors Journal, 7 (5), 707–712 (2007)
2. Starner, T., Pentland, A.: Real-time american sign language recognition from video using hidden Markov models. MIT Media Lab, Perceptual Computing Group, Cambridge, MA, Tech. Rep, 375, 265–270 (1995)
3. Mitra S., Acharya T.: Gesture recognition: A survey. IEEE Trans. Syst., Man, Cybern. C,Appl. Rev. 37 (3), 311–324 (2007)
4. Tan, U. X., et al.: Estimating Displacement of Periodic Motion with Inertial Sensors. IEEE Sensors Journal, 8 (8), 1385–1388 (2008)
5. Park, J.W., Hyun, S. D., Lee, C. W.: Real-time Finger Gesture Recognition. HCI 2008, Korea, 847–850 (2008)
6. Mäntyjärvi, J., Kela, J., Korpipää, P., Kallio, S.: Enabling fast and effortless customisation in accelerometer based gesture interaction. In: Proc. 3rd Int. Conf. Mobile Ubiquitous Multimedia, 25–31 (2004)
7. Hernandez-Rebollar, J. L., Lindeman, R. W., Kyriakopoulos N.: A Multi-Class Pattern Recognition System for Practical Finger Spelling Translation. In: Proceedings of the Fourth IEEE International Conference on Multimodal Interfaces, 185–190 (2002)
8. Ji-Hwan K., Tae-Seong K., Kyung H.: 3-D Hand Motion Tracking and Gesture Recognition Using a Data Glove. IEEE International Symposium on Industrial Electronics (ISlE 2009), Seoul Olympic Parktel (2009)
9. Dipietro, L., Sabatini, A. M., Dario, P.: A Survey of Glove-Based Systems and Their Applications. IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews, 38 (4), 461–482 (2008)
10. Swee T. T., et. Al.: Malay Sign Language Gesture Recognition system. International Conference on Intelligent and Advanced Systems (2007)

11. Zhang, X., et al.: A Framework for Hand Gesture Recognition Based on Accelerometer and EMG Sensors. IEEE Transactions on Systems, Man, And Cybernetics-Part A: Systems and Humans, 41(6), 1064–1076 (2011)
12. Platt, J.: Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines. CiteSeerX 10.1.1.43.4376 (1998)

ISSN 1870-4069

Método de análisis semántico basado en WordNet para la extracción de información en mapas conceptuales

Wenny Hojas-Mazo, Alfredo Simón-Cuevas, Manuel de la Iglesia-Campos

Universidad Tecnológica de La Habana José Antonio Echeverría, Cujae, La Habana, Cuba

{whojas, asimon, miglesia}@ceis.cujae.edu.cu

Resumen. El mapa conceptual es un tipo de representación de conocimiento basada en grafo, donde el conocimiento se expresa en lenguaje natural a través de conceptos y relaciones entre ellos. Este se puede construir manualmente o automáticamente a partir de un texto, propiciando la obtención de repositorios de conocimiento de gran valor para la gestión de conocimiento y el análisis de textos. En este trabajo, se propone un método de análisis semántico a ser aplicado en el procesamiento de consultas sobre un repositorio de mapas conceptuales. El método se basa en la extensión semántica de conceptos usando WordNet y un algoritmo de desambiguación, e incluye reglas que guían los procesos de búsqueda e integración de información. Mediante el desarrollo de un caso de estudio en el ámbito de la ingeniería ontológica se ejemplifica la ejecución del método y se muestran sus beneficios para extraer información en este tipo de repositorios.

Palabras clave: Mapas conceptuales, extracción de información, desambiguación, WordNet.

Method of Semantic Analysis Based on WordNet for Information Extraction in Concept Maps

Abstract. Concept map is a graph-based knowledge representation, where the knowledge is expressed in natural language through concepts and relationships among them. This can be built manually or automatically from texts, propitiating the obtaining of knowledge repositories with great value for the knowledge management and the texts analysis. In this work, a method of semantic analysis to be applied in the processing of queries on a concept maps repository is proposed. The method is based on the semantic extension of concepts using WordNet and a disambiguation algorithm, and several rules for guiding the search and integration of information processes are included. Through the development of a study case in the context of ontology engineering the execution of the method was exemplified and its benefits for the extraction of information in this type of repositories were shown.

Keywords. Concept maps, information extraction, disambiguation, WordNet.

1. Introducción

Los Mapas Conceptuales (MC) constituyen simultáneamente, un método para captar lo más significativo de un tema y un recurso esquemático para representar un conjunto de significados conceptuales mediante una estructura de proposiciones [1]. En los MC el conocimiento es expresado en lenguaje natural y estructurado en forma de grafo, donde los nodos representan conceptos y estos se relacionan mediante arcos dirigidos y etiquetados por una frase de enlace formando proposiciones. El MC surgen en el área de la pedagogía, pero su uso se ha extendido a otras áreas tales como: la gestión del conocimiento [2], la ingeniería ontológica [2, 3], y el análisis de textos [4].

Los MC generalmente son construidos manualmente, con la asistencia de herramientas como *CmapTools* [5], pero también se reportan propuestas para generarlos de manera automática a partir de textos [6]. Estas herramientas y métodos propician la creación de repositorios de MC (RMC), los que son reconocidos como modelos de conocimiento [5], cuando el conocimiento representado está asociado a un dominio específico. El conocimiento almacenado en los RMC puede resultar de gran valor en los diferentes contextos de aplicación de los MC, por lo que el desarrollo de soluciones que propicien el incremento de su aprovechamiento constituye un aspecto de gran interés en este ámbito. En este sentido, mejorar los mecanismos de consulta sobre este tipo de base de conocimiento es una de las metas a alcanzar para lograr ese propósito. La mayoría de las propuestas actuales que reportan mecanismos de consultas sobre RMC se centran en obtener conceptos, proposiciones, MC o estructuras proposicionales frecuentes [4, 5, 7, 8, 9, 10]. Por otra parte, en [2] se reporta CMQL (*Concept Maps Query Language*) como una propuesta más abarcadora para consultar un RMC. En CMQL se formalizan un conjunto de operaciones de consultas que facilitan obtener diferentes vistas del conocimiento almacenado en un RMC, e incluye mecanismos de integración de información como parte del procesamiento interno de cada consulta y una de sus bondades. En estas propuestas la búsqueda de información en el RMC y el procesamiento interno del conocimiento representado se lleva a cabo mediante el análisis de los conceptos a nivel sintáctico, y no a nivel semántico, donde se traten las posibles ambigüedades que puedan existir en ellos. Esto constituye una limitación para lograr un mayor aprovechamiento del conocimiento almacenado en un RMC, ya que puede provocar, por ejemplo, que no se obtenga información potencialmente útil sobre un determinado concepto debido a que no se encuentra explícitamente en el RMC, existiendo posibles conceptos que sean sinónimos. Todo lo anterior conduce a la necesidad de incorporar a los mecanismos de consultas sobre RMC soluciones que posibiliten realizar análisis semántico a nivel de conceptos, teniendo en cuenta como un elemento clave la reducción de ambigüedades. La presente investigación aborda esta problemática.

En este trabajo, se propone un método de análisis semántico dirigido a mejorar los resultados de la extracción de información a partir de consultas sobre un RMC. El método está basado en un mecanismo de extensión semántica de los conceptos del RMC a partir de *WordNet* [11], así como en la definición de un conjunto de reglas que guían la búsqueda de información e integración de información en el RMC. La extensión semántica de términos ha sido una técnica muy tratada en el ámbito de la recuperación de información para mejorar los resultados en ese proceso, pero generalmente se ha

aplicado en los términos de la consulta [12]. Esta técnica se aplica en el método propuesto sobre los conceptos incluidos en el RMC, ya que de esta manera también se contribuye a mejorar los resultados de la integración de información que se ejecuta como parte del procesamiento de las consultas. El método incluye además la aplicación de un algoritmo de desambiguación, con el objetivo de resolver las ambigüedades existentes en los conceptos y al mismo tiempo hacer más efectivo el uso de *WordNet*; siendo esta una de sus contribuciones respecto al estado del arte. La desambiguación del sentido de las palabras ha sido un tema muy abordado en ámbito de los textos, pero no así en el contexto de los MC donde se reportan pocas soluciones [13, 14], las cuales presentan algunas limitaciones. En este caso, se aplica una variante del algoritmo reportado en [14], en la cual las heurísticas se utilizan de forma combinada y no secuencialmente, con lo que se mejoran los resultados respecto a [13, 14]. El método propuesto ha sido implementado en CMQL, teniendo en cuenta el alcance de sus prestaciones para extraer información en un RMC. Mediante el desarrollo de un caso de estudio, enmarcado en el área de la ingeniería ontológica, se ejemplifica la ejecución del método, y se muestra su utilidad en la extracción de información sobre la conceptualización de una ontología.

El resto del trabajo se organiza según se describe a continuación. En la sección 2 se analizan los trabajos relacionados con la problemática tratada. En la sección 3 se presenta y describe el método de análisis semántico propuesto. En la sección 4 se describe el desarrollo de la aplicación de la propuesta en un caso de estudio y se analizan los resultados. Las conclusiones y líneas de trabajo futuro se exponen en la sección 5.

2. Trabajos relacionados

La problemática de la extracción de información en RMC se ha enfocado en la extracción de conceptos [7], proposiciones [5] o MC [8, 9], mediante procesos de búsquedas ejecutados a partir consultas definidas sobre uno o varios conceptos. En [10] se explora la viabilidad del análisis de MC a partir de la extracción de conceptos y submapas frecuentes en un RMC; reportándose algo similar en [4]. En [2] se propone el lenguaje de consulta CMQL, en el que se formalizan un conjunto de operaciones de consulta, combinando elementos de la teoría de grafos y de conjuntos, para extraer información del RMC desde diferentes perspectivas.

En esta propuesta se formalizan cuatro tipos de consultas: *unión*, *intersección*, *submap* (o *proyección*) y *extensión* [2]. En estas consultas es necesario especificar un espacio de búsqueda, conformado por MC incluidos en el RMC, y en el caso de las dos últimas, también se deben especificar conceptos de interés que guiarán la búsqueda de información a extraer. En el procesamiento interno de cada consulta se incluye un mecanismo que posibilita integrar la información contenida en el espacio de búsqueda, a partir de la identificación de conceptos que puedan ser unificados. El resultado de cada consulta de CMQL se representa en forma de MC, y este puede ser almacenado o no dentro del RMC como un nuevo MC.

Los procesos de búsqueda de conceptos en RMC que se reportan en las propuestas estudiadas, se basan fundamentalmente en la identificación conceptos incluidos en el

RMC que sean sintácticamente equivalentes a los definidos en la consulta. En el proceso de integración de información propuesto en CMQL, para determinar si dos conceptos son unificados, también se aplica este tipo de análisis sintáctico. En estas propuestas no se tiene en cuenta la posible ambigüedad presente en los conceptos incluidos en el RMC. Esto constituye una limitación para la extracción de información en RMC, ya que reduce las posibilidades de obtener información asociada a conceptos que no son sintácticamente equivalentes pero que son semánticamente similares, con respecto a los incluidos en la consulta. También el no tratar esa ambigüedad puede conducir a resultados no apropiados en el proceso de integración de información, por ejemplo: la integración de conceptos con diferentes significados. La aplicación de algoritmos de desambiguación sobre los conceptos representados en un MC puede ser considerada como parte importante de la solución a esta problemática.

La desambiguación del sentido de las palabras, conocido como *WSD* (*Word Sense Disambiguation*) es una problemática ampliamente abordada en el ámbito de los textos. Sin embargo, son pocas las soluciones reportadas a este problema en MC [13, 14] y no se ha reportado la aplicación en MC de un algoritmo de desambiguación diseñado para textos, debido fundamentalmente a las diferencias estructurales existentes entre ellos. Un MC suele constituir un resumen de un tema determinado, aunque puede representar conceptos de diferentes dominios, lo que dificulta el proceso de desambiguación, ya sea por la poca información contextual que se puede obtener sobre una palabra o concepto, o por la diversidad de dominios representados a través de sus conceptos. Un aspecto favorable es que las relaciones entre los conceptos están explícitamente representadas, lo que no ocurre en el caso de los textos y ha sido un aspecto muy bien aprovechado en las soluciones reportadas [13, 14].

En [13] se explota la topología del MC para determinar el *synset* en *WordNet* [11] que desambigua una palabra dentro de un concepto, mediante un análisis de similitud entre el contexto en *WordNet* de cada uno de los *synset* posibles (considerando solo relaciones de *hiperonimia*) y el contexto en el que se encuentra la palabra en el MC. En [14] también se incluye el análisis contextual, como una de las heurísticas para determinar el sentido de los conceptos, pero considerando también relaciones de *hiponimia*, *meronimia*/*holonimia*. También se incluyen los análisis de dominio (en correspondencia con [15]) y glosa como otras heurísticas. En esta última propuesta, el sentido se determina a partir del resultado obtenido por cada heurística, en un proceso de ejecución secuencial [14], sin considerar los beneficios que puede proporcionar la combinación de los resultado obtenidos por cada heurística.

3. Método de análisis semántico

El método incluye un mecanismo de extensión semántica de los conceptos representados en un RMC, y la definición de un conjunto de reglas que guían los procesos de búsqueda e integración de información que se llevan a cabo en el procesamiento de las consultas, específicamente de CMQL. La extensión semántica ha sido concebida con el uso de *WordNet*, y soportada en la aplicación de un algoritmo de desambiguación de conceptos inspirado en [14]. Las reglas definidas utilizan la información semántica con

la que han sido extendidos los conceptos del RMC para aumentar las capacidades del mecanismo de consulta en la identificación de información relevante a extraer, y al mismo tiempo mejorar la efectividad en el proceso de integración de información. El método puede ser aplicado para el procesamiento de MC en idioma inglés y español, requiriéndose solo el uso de la versión de *WordNet* correspondiente al idioma empleado en los MC. El RMC que se utilice como punto de partida para la ejecución del método puede incluir MC elaborados manualmente y/o generados automáticamente a partir de textos.

En la Figura 1 se esquematiza el flujo de trabajo del método propuesto instrumentado en CMQL, siendo *Q* una consulta definida formalmente por la tripleta *(I, E, t)*, donde *I* es el conjunto de conceptos sobre los que interesa extraer información, *E* el conjunto de MC del RMC que conforman el espacio de búsqueda, *t* el tipo de consulta (*u: unión, i: intersección, p: proyección, e: extensión*), E_{ext} es el conjunto de MC de *E* cuyos conceptos se han extendido semánticamente y E_{ext}* es un MC generado automáticamente en el que se integra el conocimiento representado en los MC incluidos en *E*. E_{ext} y E_{ext}* constituyen representaciones intermedias resultantes del pre-procesamiento que se realiza sobre el espacio de búsqueda.

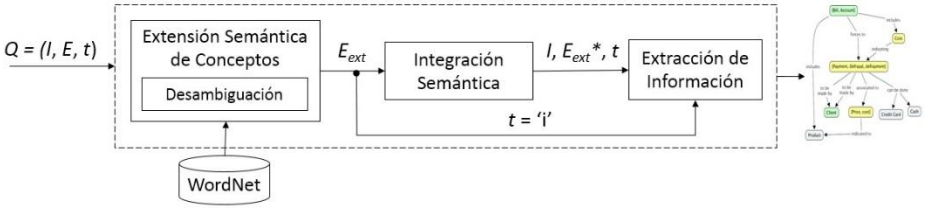

Fig. 1. Flujo de trabajo del método de análisis semántico propuesto.

3.1. Extensión semántica de conceptos

La extensión semántica de conceptos se define como el proceso de asociarle a un concepto información referente a otros términos, a partir de la identificación de relaciones de sinonimia entre ellos. En este proceso *WordNet* es utilizado como base de datos de sentidos y el mismo se aplica a todos los conceptos representados en los MC del RMC que conforman el espacio de búsqueda de la consulta. Inicialmente, se identifican y recuperan de *WordNet* todos los *synset* en los cuales están presentes los conceptos a ser extendidos. A partir de esta información los conceptos son clasificados en: ambiguos (aparece en más de un *synset*), no ambiguo (aparece en un solo *synset*) y desconocidos (no se identifican *synset* en los que está presente).

Seguidamente, se ejecuta un algoritmo de desambiguación para determinar el sentido más apropiado para cada concepto ambiguo, según el contexto en el que está siendo usado dentro del MC. En este caso, se decidió utilizar una versión mejorada del algoritmo reportado en [14], donde el sentido del concepto se determina mediante la combinación de los resultados obtenidos por cada una de las heurísticas usando una votación ponderada [16]; en correspondencia con estrategias propuestas en [17]. En experimentos parciales realizados se comprobó que esta nueva versión mejora los resultados

de [13, 14], obteniéndose valores de precisión y cobertura superiores al 80% y al 90|%, respectivamente, tanto en MC en español, como en inglés. Luego de ejecutado este algoritmo, se actualiza la lista de conceptos ambiguos y no ambiguos, ya que es posible que no todos los conceptos ambiguos puedan ser desambiguados. Los conceptos no desambiguados tendrían asociados más de un *synset*, en correspondencia con el resultado obtenido por el algoritmo de desambiguación. Al final, todos los conceptos presentes en *WordNet* son extendidos información asociada a los *synset* identificados en el proceso de desambiguación. Esta información consta de: identificador del *synset* y la lista de sinónimos.

3.2. Integración semántica del espacio de búsqueda

En este proceso se utiliza la información resultante de la extensión de conceptos para identificar posibles estructuras proposicionales que puedan ser integradas a partir de la unificación de conceptos representados en diferentes MC del espacio de búsqueda y se ejecuta como parte del procesamiento interno de cada una de las consultas de CMQL. La unificación de dos conceptos se lleva a cabo básicamente a partir de la identificación de una relación de sinonimia entre ellos y siguiendo un conjunto de reglas definidas para este propósito, las que a continuación se presentan.

Sean,

CNA: el conjunto de conceptos no ambiguos;

CA: el conjunto de conceptos ambiguos;

S(c): el conjunto de *synset* (*s*) asociados a un concepto c_i;

c_1 y c_2: dos conceptos incluidos en diferentes MC del espacio de búsqueda;

Entonces c_1 y c_2 son unificados si:

R1: $(c_1,c_2 \in CNA) \wedge (S(c_1) = S(c_2))$; o

R2: $(c_1,c_2 \in CA) \wedge (\exists s' / s' \in S(c_1) \wedge s' \in S(c_2))$; o

R3: $((c_1 \in CNA \wedge c_2 \in CA) \vee (c_1 \in CA \wedge c_2 \in CNA)) \wedge (\exists s' / s' \in S(c_1) \wedge s' \in S(c_2))$.

La etiqueta correspondiente al concepto unificado que se mostrará como resultado de la consulta se construye concatenado las etiquetas de los conceptos originales, separadas cada una por una coma ',' y encerrándolas entre corchetes '[ ]' en caso de ser más de una. Adicionalmente, se determina el *synset* que finalmente queda asociado a ese concepto unificado mediante las siguientes reglas:

R4: Si se dispara *R1*, entonces se asocia el mismo *synset* de los conceptos;

R5: Si se dispara *R2*, entonces se asocia el *synset* común a ambos conceptos;

R6: Si se dispara *R3*, entonces se asocia el *synset* correspondiente al $c_i \in CNA$.

3.3. Extracción de información

Este proceso se centra fundamentalmente en la evaluación de la consulta *Q*, teniendo en cuenta su tipo y los elementos que la definen. En la especificación de la consulta *unión* no se definen conceptos de interés (por tanto, $I = \{\}$), y se obtiene como resultado un nuevo MC que representa una vista integrada de todos los MC que componen en el espacio de búsqueda, obteniéndose E_{ext}*. Este tipo de consulta se puede ejecutar de

manera directa por un usuario, pero también es usada como paso intermedio en la ejecución de otras consultas. En la consulta *intersección* tampoco son definidos conceptos de interés, y tiene el propósito de identificar y extraer aquellos conceptos y proposiciones que son comunes a todos o a un subconjunto de los MC incluidos en E_{ext}.

Las consultas *proyección* y *extensión*, están concebidas para extraer del RMC estructuras proposiciones que muestren información relacionada con los conceptos de interés de la consulta (*I*). Por tanto, su procesamiento resulta ser más compleja porque en el proceso búsqueda hay que determinar a partir de qué conceptos (de los incluidos en E_{ext}) es que se extrae la información requerida. La identificación de los conceptos de interés en E_{ext} se lleva a cabo mediante las reglas que a continuación se presentan, las cuales son comprobadas en el mismo orden en el que han sido numeradas.

Sea *a* un concepto incluido en la consulta tal que $a \in I$, *b* un concepto representado en algún mc_i tal que $mc_i \in E_{ext}$, $Pl(c_i)$ la lista de palabras que componen la etiqueta de un concepto c_i, y $si(c_i)$ el conjunto de sinónimos asociados a c_i.

Entonces, *b* es identificado como concepto de interés si:

R1: *a* y *b* son sintácticamente equivalentes; o

R2: $a \in Pl(b)$; o

R3: $a \in si(b)$;

4. Aplicación en caso de estudio

El caso de estudio se enmarca en el área de la ingeniería ontológica, en la cual se investiga sobre principios, métodos, y herramientas para la construcción y mantenimiento de ontologías [18]. La obtención de una descripción informal o conceptualización del conocimiento que se quiere formalizar en la ontología, usualmente representada mediante conceptos y relaciones entre ellos, constituye una tarea común en las etapas tempranas de la mayoría de las metodologías existentes para la construcción de ontologías.

En este contexto, la utilidad de los MC para capturar y representar ese tipo de conceptualizaciones ha sido reconocida por varios autores [2, 3]. El MC facilita la captura del conocimiento que poseen los expertos del dominio, por la flexibilidad que brindan para representar el conocimiento y porque constituyen una herramienta muy intuitiva para las personas. El MC también ayuda al ingeniero de conocimiento a identificar los conceptos más significativos del dominio y los diferentes tipos de relaciones que se pueden establecer entre ellos en ese contexto.

En las Figuras 2 y 3 se muestran dos MC en idioma inglés, nombrados 'Cuenta' y 'Pago' respectivamente, elaborados manualmente usando *CmapTools* y que constituyen fragmentos de una conceptualización del dominio de gestión hotelera, construida como resultado de la captura de conocimiento de expertos. La obtención de esta conceptualización constituyó uno de los primeros pasos llevados a cabo en la construcción de una ontología terminológica que sería empleada para la extensión semántica de consultas en un sistema de recuperación de información. La ontología no solo debía incluir conceptos asociados a ese dominio específico, los cuales se representarían como clases, sino también debía incluir sinónimos asociados a dichos conceptos.

Los objetivos del desarrollo de este caso de estudio están dirigidos a ejemplificar la aplicación del método propuesto y a mostrar su utilidad para la extracción de información sobre los MC que se muestran en las Figuras 2 y 3, como parte del análisis que puede realizar el ingeniero de conocimiento para la obtención de la ontología. En este sentido, se describe el procesamiento que se lleva a cabo como parte de la ejecución de consultas que responden a los siguientes requisitos informacionales del ingeniero de conocimiento:

1. Obtener una vista global de la conceptualización capturada;
2. Obtener información sobre los conceptos específicos *'defrayment'* y *'cost'*.

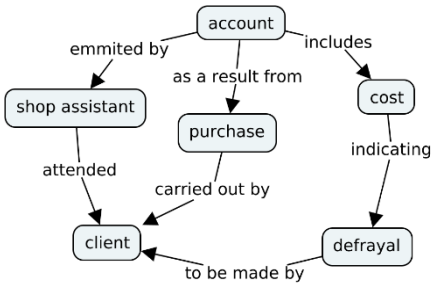

Fig. 2. Mapa conceptual 'Cuenta'.

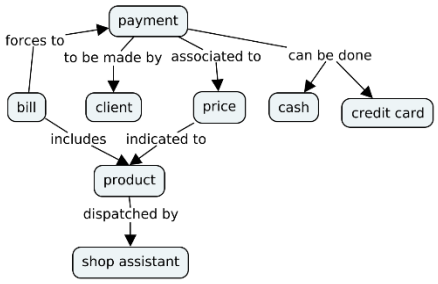

Fig. 3. Mapa conceptual 'Pago'.

A través de CMQL, el primer requisito se puede satisfacer mediante la ejecución de la consulta *unión*, cuya formalización sería *UM({Cuenta, Pago})*, y el segundo requisito se puede satisfacer ejecutando una consulta *proyección*, cuya formalización sería SM^{+1}*({Cueta, Pago}, {'defrayment', 'cost'}),* según [2].

El primer paso que se ejecuta en cada consulta, como parte del método, es la extensión semántica de los conceptos representados en los MC *'Cuena'* y *'Pago'*. En este proceso inicialmente se identifican los *synset* en *WordNet* asociados a cada concepto y luego se trata de eliminar las ambigüedades con el algoritmo de desambiguación propuesto, cuyos resultados se muestran en las Tablas 1 y 2. Se puede apreciar en estas tablas el alto grado de ambigüedad de los conceptos, teniendo en cuenta que como promedio la cantidad de *synsets* de *WordNet* asociados a dichos conceptos en resultó ser de 5 *synsets* en ambos MC.

Tabla 1. Resultados del algoritmo de desambiguación en 'Cuenta'.

Conceptos	***Synsets***	**Sentidos identificados**
account	*13*	*06516955 - Economy - {bill,* ***account,*** *invoice}*
purchase	*5*	*00079018 - Economy - {***purchase***}*
shop assistant	*1*	*10548227 - Commerce - {salesclerk, shop_clerk, clerk,* ***shop_assistant****}*
client	*3*	*09984659 - Commerce - {customer,* ***client****}*
cost	*5*	*05163807 - Factotum - {price,* ***cost****, toll}*
defrayal	*3*	*01120448 - Economy - {payment,* ***defrayal****, defrayment}*

La cobertura del algoritmo de desambiguación en ambos MC fue del 100 %, e igual resultado de precisión se obtuvo en 'Pago'. En el caso de 'Cuenta' la precisión fue del 83 % ya que uno de los conceptos no se logró desambiguar satisfactoriamente, específicamente el concepto *'cost'*. Estos resultados se pueden considerar prometedores, teniendo en cuenta que los MC son pequeños, por tanto, con poca información contextual a tener en cuenta en el análisis, y con alto grado de ambigüedad de los conceptos representados.

Tabla 2. Resultados del algoritmo de desambiguación en 'Pago'.

Conceptos	***Synsets***	**Sentidos identificados**
payment	*3*	*01120448 - Economy - {**payment**, defrayal, defrayment}*
bill	*13*	*06516955 - Economy - {**bill**, account, invoice}*
credit card	*1*	*13376012 - Banking - {**credit_card**, charge_card, charge_plate, plastic}*
cash	*4*	*13386614 - Money - {**cash**, hard_cash, hard_currency}*
price	*9*	*05145118 - Money - {monetary_value, **price**, cost}*
client	*3*	*09984659 - Commerce - {customer, **client**}*
shop assistant	*1*	*10548227 - Commerce - {salesclerk, shop_clerk, clerk, **shop_assistant**}*
product	*6*	*03748886 ' Commerce - {merchandise, ware, **product** }*

Finalmente, como resultado de la extensión semántica cada uno de los conceptos representados en los dos MC fueron extendidos con el identificador del *synset* identificados y los términos (sinónimos) incluidos en ellos, y que se muestran en las tablas anteriores. Luego de obtener los MC extendidos (E_{ext}), se lleva a cabo el proceso de integración semánticas, como parte también de la ejecución cada consulta. Con el objetivo de mostrar y analizar la contribución del método para la extracción de información sobre los MC seleccionados, se decidió ejecutar cada consulta siguiendo dos variantes: (1) sin usar el método de análisis semántico y (2) usando el método análisis semántico. Los resultados de la consulta *unión* (en sus dos variantes) se muestran en las Figuras 4 y 5, y los de la consulta *proyección* se muestran en las Figuras 6 y 7. En las Figuras 5 y 7 también se pueden apreciar resultados de la unificación de conceptos.

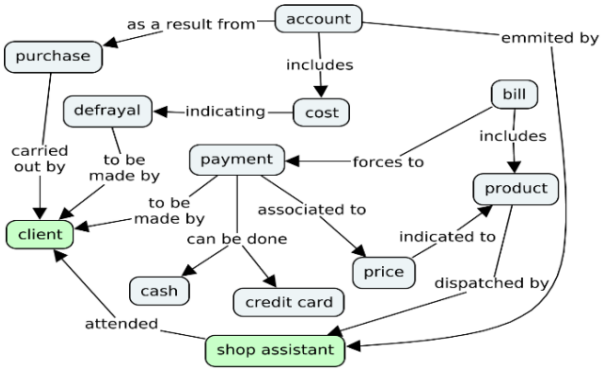

Fig. 4. Resultado de la *unión* sin usar el método.

En la Figura 4 se aprecia que ambos MC se integraron solo a partir de los conceptos *'client'* y *'shop assistance'*, quedando representados como nodos diferentes los conceptos *'account'* y *'bill'* que tienen el mismo sentido. Esto se resuelve con la aplicación del método propuesto, como se muestra en la Figura 5. En esa figura se puede apreciar como son unificados aquellos conceptos con igual significado, según los resultados de la desambiguación, tal es el caso de: [*defrayal, payment*], que se corresponden a conceptos sintácticamente diferentes, así como '*client'* y *'shop assistance',* que se corresponden a conceptos sintácticamente equivalentes.

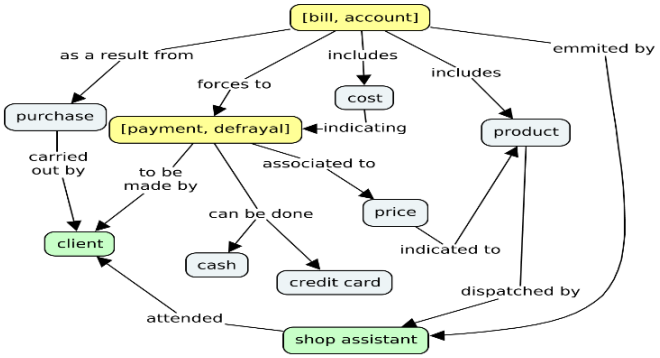

Fig. 5. Resultado de la consulta *unión* usando el método.

Este resultado muestra que con la aplicación del método se alcanza una mayor precisión con respecto a la información extraída del RMC, respecto a propuestas que no tratan la semántica. Además, también aporta beneficios para el ingeniero de conocimiento, a la hora de llevar a cabo la etapa de formalización de la ontología. Por ejemplo, el resultado mostrado en la Figura 5 facilita identificar grupos de conceptos semánticamente similares (conceptos unificados) que deben ser codificados como una única clase en la ontología, y suministra información sobre posibles sinónimos de dicha clase.

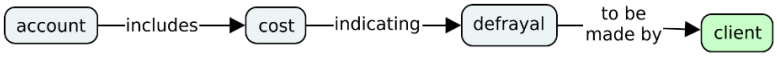

Fig. 6. Resultado de la *proyección* sobre los conceptos *'defrayment'* y *'cost'* sin usar el método.

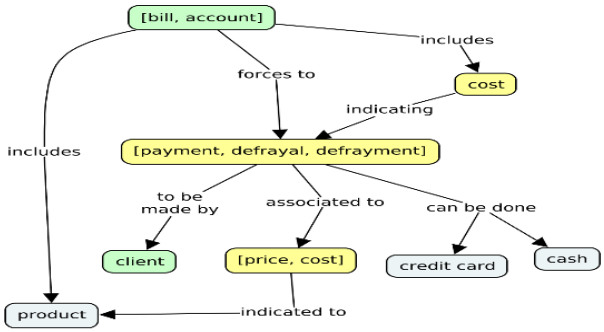

Fig. 7. Resultado de la *proyección* sobre los conceptos *'defrayment'* y *'cost'* usando el método.

En la Figura 7 se puede apreciar como aumenta la cantidad de información (conceptos y proposiciones) extraída y vinculada a los conceptos de la consulta al aplicar el método propuesto, respecto a lo mostrado en la Figura 6; siendo esta una de sus contribuciones. A partir de la aplicación del método fue posible extraer información sobre el concepto *'defrayment'*, no estando representado explícitamente en ninguno de los dos MC, ya que se identificó que existían conceptos que tenían una relación de sinonimia con ese término. Por otra parte, este resultado también es de utilidad para el ingeniero de conocimiento, ya que mediante la consulta realizada puede identificar si un nuevo término del dominio, en este caso *'defrayment'*, ya se encuentra representado en la conceptualización, y a partir de ello tomar decisiones sobre como incluirlo en la ontología, ya sea como una nueva clase o como un sinónimo de una clase existente.

5. Conclusiones y trabajo futuro

El análisis de la semántica asociada a los conceptos representados en un MC es esencial para su procesamiento computacional, ya que esta se encuentra implícita. En el trabajo se ha presentado un método que posibilita dotar a los mecanismos de consulta sobre RMC, y en particular a CMQL, de capacidades para realizar este tipo de análisis, sobre la base de asociar a los conceptos incluidos en el espacio de búsqueda información semántica recuperada de *WordNet*. Se aplica un algoritmo de desambiguación para reducir las ambigüedades que puedan tener algunos de esos conceptos. La información semántica asociada a los conceptos se utiliza en la definición de un conjunto de reglas que guían los procesos de búsqueda e integración de información que se ejecutan como parte de una consulta. La combinación de estos elementos permitió incrementar las capacidades de las consultas de CMQL para encontrar información potencialmente útil en el RMC, y contribuyó a mejorar la precisión en la integración de la información dentro del RMC. El desarrollo del caso de estudio posibilitó ejemplificar el funcionamiento del método, y sus resultados evidenciaron los beneficios que ofrece para la extracción de información en un RMC, específicamente en el análisis y procesamiento de la conceptualización que sirve de base para la construcción de una ontología. Se comprobó que el método posibilita incrementar de la cantidad de información a obtener a partir de una consulta y la efectividad en la integración de información.

En el futuro, se mejorarán las propuestas de extensión semántica de conceptos e integración de información definidas, aportando una solución para cuando los conceptos no estén incluidos en *WordNet*, lo que suele ocurrir cuando representan nombres de entidades o son de dominio específico. También se trabajará en el diseño de un entorno de evaluación experimental, actualmente no identificado en la bibliografía consultada, que posibiliten medir de una manera más objetiva los resultados que se obtienen con el método, y establecer líneas de comparación con propuestas similares.

Referencias

1. Novak, J. D., y Gowin, D. B.: Learning How to Learn. Cambridge University Press (1984)

2. Liu, S. H., y Lee, G. G.: Using a concept map knowledge management system to enhance the learning of biology, Comput. Educ. 68, 105–116 (2013)
3. Simón, A., L. Ceccaroni, L., Rosete, A., Suárez, A., y Victoria, R.: A support to formalize a conceptualization from a concept maps repository. In: Proc. of CMC'08, 68–75 (2008)
4. Rizzi, R., y Parente de Oliveira, J. M.: Concept maps as the first step in an ontology construction method. Information Systems, 38, 771–783 (2013)
5. Rodríguez, L., Hojas, W., y Simón, A.: Método para la identificación de submapas frecuentes en modelos de conocimiento. CICCI'16, La Habana, Cuba (2016)
6. Cañas, A. J., Hill, G., Carff, R., Niranjan, S., Lott, J., Eskridge, T. C., Gómez, G., Arroyo, M., y Carvajal, R.: Cmaptools: a knowledge modeling and sharing environment. In: Proc. of CMC'04, 1, 125–13 (2004)
7. Zubrinic, K., Kalpic, D., y Milicevic, M.: The automatic creation of concept maps from documents written using morphologically rich languages. Expert Systems with Applications, 39 (16), 12709–12718 (2012)
8. Leake, D., Maguitman, A., Reichherzer, T., Cañas, A. J., Carvalho, M., Arguedas, M., y Eskridge, T.: Googling from a concept map: Towards automatic concept map-based query formation. In: Proc. of CMC'04, 1, 409–416 (2004)
9. Cañas, A. J., Leake, D. B., y Maguitman. A. G.: Combining Concept Mapping with CBR: Towards Experience-Based Support for Knowledge Modeling. In: Proc. of FLAIRS Conference, AAAI Press, 286–290 (2001)
10. Eskridge, T. C., Granados, A., y Cañas, A. J.: Ranking concept map retrieval in the CmapTools network. In: Proc. of CMC'06, 1, 7–484 (2006)
11. Yoo, J. S., y Cho, M.-H.: Mining Concept Maps to Understand University Students' Learning. International Educational Data Mining Society, ERIC, 1, 19–21 (2012)
12. Miller, G., y Fellbaum, C.: WordNet: An Electronic Lexical Database. The MIT Press: Cambridge, MA (1998)
13. Carpineto, C., y Romano, G. A: Survey of Automatic Query Expansion in Information Retrieval. ACM Comput. Surv. 1 (44), 1–50 (2012)
14. Cañas, A., Valerio, A., Lalinde, J., Carvalho, M., y Arguedas M.: Using WordNet for Word Sense Disambiguation to Support Concept Map Construction. LNCS, Springer, 2857, 350–359 (2003)
15. Simón, A., Ceccaroni, L., Rosete, A., Suárez, A., y de la Iglesia, M.: A concept sense disambiguation algorithm for concept maps. In: In Proc. of CMC'08, 14–21 (2008)
16. Bentivogli, L.; Forner, P.; Magnini, B.; y Pianta, E.: Revising WordNet Domains Hierarchy: Semantics, Coverage, and Balancing. In: Proc. of COLING 2004 Workshop on Multilingual Linguistic Resources, 101–108 (2004)
17. Klein, D., Toutanova, K., Ilhan, H., Kamvar, S., y Manning, C.: Combining heterogeneous classifiers for word-sense disambiguation. In: Proc. of the ACL-02. 8, 74–80 (2002)
18. Navigli, R.: Word Sense Disambiguation: A Survey. ACM Computing Surveys, 41(2), 1–69 (2009)
19. Sure, Y., Staab, S., Studer, R.: Ontology Engineering Methodology. In: Staab S., Studer R. (Eds.), Handbook on Ontologies, Springer-Verlag (2009)

ISSN 1870-4069

Generación de multitudes virtuales heterogéneas basadas en patrones de agrupación de comportamiento humano

Fernando Rebollar[1], Marco A. Ramos[1], Vianney Muñoz[1], Félix F. Ramos[2]

[1] Universidad Autónoma del Estado de México (UAEMex), Facultad de Ingeniería, México

[2] Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional (CINVESTAV), Unidad Guadalajara, México

fc@live.com.mx, marco.corchado@gmail.com, vmunozj@uaemex.mx, framos@gdl.cinvestav.mx

Resumen. Las ciencias computacionales en conjunto con otras áreas del conocimiento participan en el estudio del comportamiento de masas de individuos, intentando predecir y anticipar situaciones que puedan presentarse, a través de simulaciones de multitudes que permitan una experimentación a bajo costo y que aporten datos a las organizaciones para predecir y anticipar distintas eventualidades. Para ello, es necesario realizar simulaciones con multitudes virtuales que se comporten lo más parecido a la realidad. En este artículo se presenta una forma de generar multitudes virtuales con comportamientos heterogéneos, de tal forma que los individuos que conforman la multitud tengan distintos comportamientos. Se propone una técnica de agrupamiento en distintas regiones del área a poblar utilizando diagramas de Voronoi, posibilitando la caracterización de zonas donde existe una concentración masiva de personas en un espacio específico como lo son: plazas, centros comerciales, aeropuertos, escuelas, etc.

Palabras clave: Inteligencia artificial, agentes, multitudes heterogéneas.

Generation of Heterogeneous Virtual Crowds based on Patterns of Grouping of Human Behavior

Resumen. Computer science with other areas of knowledge is involved in the study of the individual's behavior of the crowd, trying to predict and anticipate situations that may arise. In crowd simulations, we can see differents experimentations that provide the different behavior on specific conditions or eventualities not controlled. In this paper, we present one technique based on Voronoi diagrams to defined the local concentration of people (LCP) that permitted create the heterogeneous crowd. LCP

identified the principal place of environment where to exist the possibility of a crowd. The Voronoi diagrams help to agents to find the particular location to access a resource to complete the main goal. The first results showed the way of how the agents can crowd like the circles, delta, and oval groupping.

Keywords. Artificial intelligence, agents, heterogeneous crowds.

1. Introducción

El estudio de la concentración de individuos en lugares públicos como plazas, centros comerciales, parques, jardines, etc., es un campo de estudio abierto en las diferentes disciplinas de las ciencias. Conllevando a la necesidad de disponer de sistemas que permitan pronosticar y predecir eventualidades en situaciones no controladas, como es en el caso de un terremoto. De ahí que, las ciencias computacionales como la inteligencia artificial investigan cómo replicar el comportamiento humano virtualmente, para obtener simulaciones que sean lo más apegado a la realidad, apoyándose de las áreas sociales, psicológicas, neurocientíficas, entre otras, con la finalidad de empatar estas teorías en el área de la inteligencia artificial.

El principal propósito de estudiar y simular multitudes virtuales es representar con precisión grupos de individuos autónomos llamados agentes virtuales que atienden a las mismas reglas en entornos cercanos a aquellos encontrados en la vida real. Para anticipar contingencias y atenderlas adecuadamente antes de que sucedan es necesario estudiar el comportamiento de las masas de individuos en actividades cotidianas, permitiendo la construcción de la infraestructura requerida de acuerdo a las necesidades de los individuos que hacen uso de ella, por ejemplo, las plazas, los aeropuertos, las escuelas, el transporte público, etc. A esta infraestructura se le conoce como ciudades inteligentes.

Grupos de investigación alrededor del mundo se han sumado al problema de la simulación de multitudes virtuales con grandes logros desde el realismo obtenido hasta el comportamiento embebido en los agentes virtuales. Sin embargo, sus simulaciones carecen de heterogeneidad en la población ya que sus estudios se enfocan en un sólo modelo que replican infinitamente en el medio ambiente virtual, esto resulta menos costoso computacionalmente comparado a la generación diversos modelos de acuerdo a la heterogeneidad de la población e implica que el comportamiento observado dentro de la simulación sea idéntico para todos los agentes virtuales [4].

En la realidad no existe un número determinado de individuos que constituyen una multitud, no obstante, el estudio de las multitudes se basa en el mayor número de individuos concentrados en un espacio observando su comportamiento en situaciones de estrés, manteniendo la meta de poder anticipar contingencias en tiempo real [11]. Las aproximaciones computacionales para la generación de multitudes son centradas en las reglas propias del ambiente físico y el comportamiento está basado en la interacción del ambiente y las reglas físicas del mismo.

Uno de los problemas que se tienen con la generación de multitudes es la diversidad visual para representar la heterogeneidad en las multitudes. Así como la planificación que deberán de realizar los agentes dentro del ambiente virtual para el logro de sus objetivos [22].

Para reproducir comportamientos individuales y grupales de forma exitosa dos problemas fundamentales deben abordarse: la planificación y la toma de decisiones. La planificación dotará a los agentes para poder observar el medio ambiente y poder decidir sobre las condiciones del mismo, por ejemplo la evasión de colisiones. Los métodos principales para la evasión de colisiones se basan en fuerzas sociales [9] mientras que la velocidad es manejada de manera recíproca para todos los agentes. Un algoritmo simple para evitar colisiones compara la posición de cada agente con los demás, sin embargo conforme el número de agentes (denotado por N) crece, la complejidad del algoritmo es $O(N^2)$. Dicha complejidad representa un problema cuando se requiere simular grandes multitudes e imposibilita la obtención de una simulación en tiempo real. Para que una simulación de multitudes sea precisa es necesario que reproduzca comportamientos humanos individuales y grupales, además los algoritmos que sintetizan estos comportamientos deben estar optimizados para trabajar en tiempo real [17].

El comportamiento colectivo en los humanos se ha estudiado desde principios del siglo XIX con la finalidad de observar las acciones que las personas realizan cuando se presentan diferentes circunstancias como celebraciones, manifestaciones, simulacros o incluso cuando ocurren fenómenos naturales como incendios o terremotos [6]. Las ciencias computacionales realizan esfuerzos para simular de manera virtual multitudes con el objetivo de reproducir comportamientos similares a los de los seres humanos con la finalidad de estudiar eventualidades en las multitudes [4].

Las simulaciones de multitudes son realizadas a partir de diferentes enfoques dependiendo del ambiente virtual que se desea poblar, por ejemplo: la industria del entretenimiento (videojuegos, películas, realidad virtual, etc.). Este tipo de simulaciones ha tomado importancia en el uso de los llamados *juegos serios* [16]. Estos últimos requieren ser poblados con agentes que permitan al usuario contar con una retroalimentación o ser asistido en tareas inmersas en el ambiente virtual.

Fig. 1. Multitud de personas en avenidas públicas.

Este tipo de simulaciones permite a las diferentes organizaciones garantizar la integridad de los individuos aglomerados en espacios públicos, como se ilustra en la Figura 1. Las organizaciones hacen uso de las simulaciones basadas en esté tipo con el propósito de prevenir situaciones no controladas en el entorno [25].

2. Trabajos relacionados

El estudio de las multitudes es de suma importancia y se puede observar ante la presencia de fenómenos naturales, por ejemplo los cardumen de sardinas, en donde las especies jóvenes se encuentran en el centro para protegerse de los predadores y preservar la especie. En 1987 Craig Reynolds presentó uno de los primeros trabajos sobre la creación de multitudes basados en el comportamiento de las aves y la manera de cómo se agrupan en vuelo. Las investigaciones de Reynolds encaminan a la generación de multitudes basadas en humanoides siguiendo tres reglas simples: separación, alineación y cohesión. Según Reynolds un ave es consciente de tres elementos durante el vuelo: conocimiento de sí mismo, vecinos cercanos, y un líder a seguir [15].

Los trabajos de Reynolds funcionan en agentes que tienen un mismo comportamiento como son las aves y otro tipo de animales. Sin embargo, las multitudes basadas en humanos es más compleja debido a la personalidad de cada uno de los individuos que participan en una multitud, donde diversos comportamientos deben de ser considerados para que la simulación se apegue más a la realidad.

Es importante considerar en una simulación de multitud un comportamiento reactivo, es decir asociar todas aquellas acciones que son desencadenadas mediante un evento, en tanto no se registre ningún estimulo del ambiente o incluso de algún otro agente involucrado en la simulación. Los agentes que pertenezcan a la categoría reactiva no tienen ninguna razón para reaccionar [4]. Erik Millan genera máquinas de estado finitas desde archivos XML y las guarda en imágenes que los agentes pueden consultar [11]. Pelechano y Badler han combinado reglas de percepción y comportamientos reactivos para dirigir agentes en entornos virtuales [13]. Sí los comportamientos reactivos son implementados correctamente pueden tener como consecuencia comportamientos emergentes que producen simulaciones más realistas.

Las simulaciones basadas en reglas no requieren de un razonamiento complejo, esto es debido a que el medio ambiente es el responsable de seleccionar la mejor acción que los agentes deben seguir en las diferentes situaciones que se les presentan [20]. Sin embargo, esto se aleja a lo que sucede en un entorno real porque el ambiente no controla las acciones de los peatones, sólo los limita. Kapadia [10], combinó predicciones de espacio-tiempo, comportamientos reactivos, y movimientos de dirección en plataformas dedicadas a la simulación de multitudes, lo que da libertad a los agentes de tomar sus decisiones de desplazamiento por ellos mismos.

Lograr simulaciones de multitudes lo más parecidas a la realidad, requiere de resolver el problema de coaliciones que medido computacionalmente es de $O(N^2)$, donde N es el número de agentes que participaran en la simulación [17].

Esto significa que cada agente tiene que consultar a todos los otros agentes por su posición y otra información importante que le permita calcular su dirección y velocidad con respecto a la de sus vecinos. Esto reduce la complejidad de las búsquedas de proximidad, permitiendo simulaciones de grandes multitudes.

Para reducir la complejidad en el paso de mensajes entre los agentes se utilizan estructuras jerárquicas, como los *octrees*, en donde el espacio se subdivide en varias regiones que contienen agentes donde los integrantes de una región conocen solo a los agentes que estén dentro de la misma región [8]. Mejoras a la estructura *octrees*, utilizan *árboles kd* para que al consultar a los vecinos más cercanos sea un proceso eficiente. Bleiweiss en [1] presentó una implementación en paralelo de la biblioteca popular para evasión de colisiones, obteniendo un aumento de velocidad de 4.8X en comparación con la implementación original. Bleiweiss cambio el método de búsquedas de proximidad de un *árboles kd* a un método basado en tablas *hash* con el fin de mejorar el rendimiento en la Unidad de Procesamiento Gráfico (GPU).

En 2004, Chenney [2] propone una técnica para la representación y el diseño de campos de velocidad, usando autómatas celulares útiles para crear movimientos de flujo, que son seguidos por los peatones, con el fin de moverse a través de un entorno, ver la Figura 2. En 2011, Zhang [24] presenta un modelo en el que las celdas de un autómata celular representan posiciones discretas en el espacio, utilizadas por los peatones para moverse al cambiar de una celda a otra generando simulaciones más realistas.

Fig. 2. Simulación de Chenney usando autómatas celulares [2].

Guy en 2010, presenta un método que calcula trayectorias reduciendo al mínimo el esfuerzo que los agentes necesitan llevar a cabo para llegar a su destino [7]. El algoritmo es capaz de evitar colisiones con otros agentes y los obstáculos, al tiempo que permite simulaciones en tiempo real [12].

Van den Berg mejora el algoritmo de RVO (Velocidades Reciprocas para Obstáculos) reduciendo el problema a un programa lineal de baja dimensión [23], y con ello son capaces de simular multitudes de miles de personajes. En la Figura 3 se observa una prueba con 1,000 agentes en tiempo real.

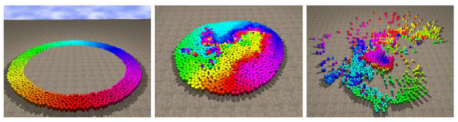

Fig. 3. Los agentes se mueven hacia la posición opuesta de donde comienzan en el círculo.

D. Thalmann [21], remarca que las principales razones que dificultan el uso de multitudes virtuales es precisamente el dominio de tiempo real y las altas exigencias al CPU, así como los altos costos de la producción de contenidos. Thalmann propone algoritmos para optimizar el hardware haciendo posible mostrar escenas virtuales en 3D con miles de entidades individuales animadas que anteriormente no era posible, su simulador permite crear miles de agentes donde la multitud se mueve de un lugar a otro en tiempo real (ver Figura 4).

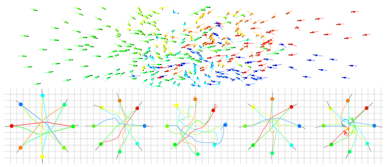

Fig. 4. Simulación en tiempo real de Thalmann [14].

3. Generación de comportamientos heterogéneos

Un problema a abordar en la generación de multitudes es la carencia de diversidad que la conforman, debido a que sólo se utiliza un modelo del agente que se replica de manera discriminada, cambiando sólo el color de la ropa o cabello. También podemos observar la falta de autonomía en los agentes provocando un solo comportamiento lo cual reduce el realismo en la simulación.

Nuestra propuesta se basa en la generación de multitudes heterogéneas a través de diversos modelos que conforman la multitud sin descuidar los comportamientos asociados a cada agente. Por ejemplo, sí en la escena se observa un anciano este debe tener el comportamiento de un anciano.

3.1. El comportamiento humano

El comportamiento de cada persona en la vida real es único, el cual esta determinado por un sin fin de factores que lo van determinando a lo largo del tiempo y de experiencias vividas por cada persona. También lo delimitan aspectos fisiológicos y capacidades que impiden poder realizar algunas actividades para determinados sectores de la población.

Lo ideal en las simulaciones de multitudes es que cada agente se comportara de manera diferente tal y como pasa en la vida real. Sin embargo, este proceso tomaría demasiados recursos del sistema para que cada agente virtual contara con un comportamiento individual impidiendo su simulación en tiempo real.

Samson en [19] determinó la velocidad de desplazamiento, cadencia de los pasos y longitud de zancada de las personas, considerando parámetros como la edad, el peso y la altura de las personas. Para ello, analizó a 118 mujeres y 121 hombres en un rango de entre los 19 a 90 años de edad, los cuales caminaban a su velocidad preferida como lo harían normalmente a través de una pasarela de 12 metros. Samson propone formulas para calcular la velocidad de desplazamiento de los individuos en actividades cotidianas. En nuestro caso de estudio retomamos las ecuaciones propuestas por Samson para que los agentes virtuales reproduzcan estos desplazamientos con la característica que podemos distinguir hombres y mujeres dentro de la simulación. La Tabla 1 muestra la relación de velocidad desplazamiento existente entre hombres y mujeres.

Tabla 1. Ecuaciones de velocidades de desplazamiento en metros/segundos de los seres humanos de Samson [19].

Hombres
Velocidad = 1.460
Velocidad = -0.002 edad (*) + 1.582
Velocidad = -0.002 edad (*) + 0.442 altura (*) + 0.750
Velocidad = -0.001 edad (*) + 0.486 altura (*) - 0.001 peso (*) + 0.720
Mujeres
Velocidad = 1.420
Velocidad = -0.003 edad (*) + 1.552
Velocidad = -0.002 edad (*) + 0.618 altura (*) + 0.484
Velocidad = -0.001 edad (*) + 0.827 altura (*) - 0.003 peso (*) + 0.316

La posibilidad de dotar a los agentes de valores aleatorios en el desplazamiento, así como condicionantes de masa y altura permite obtener una simulación dentro de la multitud lo más cercana a la realidad. Los factores asociados a la física del ambiente son comportamientos emergentes de acuerdo a las diferentes masas de los agentes que participan en la simulación. Los agentes están provistos de sensores que les permiten recalcular sus trayectorias para evitar colisionar con objetos dentro de ambiente y con otros agentes. En un sistema multiagentes el ambiente se define como el conjunto $E = \{e, e', ...\}$ donde E es el ambiente conformado por todos los posibles estados. $Ac = \{\alpha, \alpha', ...\}$ representa todas las acciones permitidas dentro de E. Los agentes construyen su base de conocimientos a partir de $r : e_0 \overset{\alpha_0}{\rightarrow} e_1 \overset{\alpha_1}{\rightarrow} e_2 \overset{\alpha_2}{\rightarrow} e_3 \overset{\alpha_3}{\rightarrow} \cdots \overset{\alpha_{u-1}}{\rightarrow} e_u$ que representan las acciones realizadas de un estado a otro dentro de E.

3.2. Distribución geométrica espacial del ambiente

La geografía del ambiente es un factor importante en la generación de multitudes, los agentes virtuales que ocuparán el espacio deberán contar con información inicial que les permita reconocer su entorno y poder lograr sus objetivos, así mismo saber qué lugares son los que pueden visitar y los que deben de evitar. Para resolver esto, en nuestro caso hacemos uso de los diagramas de Voronoi que permiten dividir un área en regiones bien definidas.

Sea $P = \{p_1, p_2, ..., p_n\}$ un conjunto de puntos en el plano, haciendo uso del diagrama de Voronoi es posible asignar a cada punto una región en el plano correspondiente a los puntos más cercanos, una región para cada $p_i \in P$, todos los puntos asignados a p_i en el conjunto de puntos P forman la región de Voronoi $V(p_i)$ [3]. Dado un conjunto de puntos en P y un punto de consulta q, es posible determinar el punto más cercano a q en P, dado que la ubicación de q esta dentro de una región de Voronoi en un punto p_i la cual indica que dicho punto p_i es el más cercano al punto q.

$$V(p_i) = q|\|p_iq\|| < \|p_jq\|, \forall j \neq i, \tag{1}$$

donde $\|pq\|$ es la distancia euclídea entre p y q.

Como primer aproximación podemos decir que dado un conjunto P de sitios (puntos) en el plano, su diagrama de Voronoi es la partición de ese plano en regiones (una región para cada sitio), tal que la región del sitio p contiene todos los puntos del plano que están más cerca de p que de cualquier otro sitio en P.

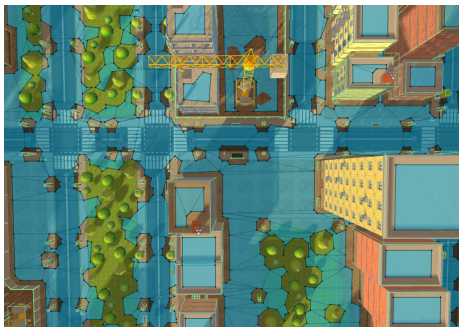

Fig. 5. Ubicación de LCP dentro del ambiente.

Nuestro caso de estudio hace uso del algoritmo Steven Fortune para generar el diagrama de Voronoi ya que dicho algoritmo [5] se ejecuta en $O(n \log n)$ por lo que se puede notar que es un algoritmo en $O(n)$ considerado uno de los mejores actualmente. Una vez generado el diagrama de Voronoi es utilizado para determinar la correcta distribución de la población dentro de ambiente virtual, para los primeros resultados trabajaremos con una ciudadela simulada

(ver Figura 5), donde los puntos p generados por Voronoi los llamaremos lugares de concentración de población (LCP). Un LCP puede ser una parada de autobús, una tienda comercial, etc. El uso de LCP nos permite conceptualizar los espacios dentro del ambiente y poder definir la concentración de individuos por regiones, por ejemplo si la región esta marcada como escuela primaria, dicha región deberá ser poblada en su mayoría de agentes que representen niños y niñas en un rango de edad entre los 6 y 12 años.

La posibilidad de contar con varios LCP nos permitirá poblar ambientes mucho más complejos y así poder observar y estudiar los comportamientos asociados y emergentes dentro de una población completamente heterogénea.

4. Resultados

El software utilizado para las simulaciones de los agentes fue Unity 5.3 de 64 bits, en una computadora con procesador intel i7 de 3.4GHz y 4 GB en RAM. Los primeros resultados obtenidos se basan en los desplazamientos que toman los agentes dentro del ambiente virtual en donde es posible distinguir desplazamientos heterogéneos, inicialmente los agentes están representados por cubos de diferentes tamaños y masas para dotarlos de características individuales.

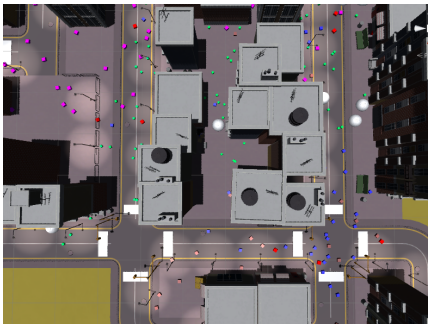

Fig. 6. Agentes virtuales, desplazándoce por la ciudad buscando su LCP de interés.

Los agentes son identificados por colores para clasificarlos en 5 tipos de roles dentro del ambiente. Los de color verde representan a la población de niños entre 4 a 14 años de edad, los de color azul representan a jóvenes de 15 a 24 años, los rojos representan a adultos de 25 a 54 años de edad, los violeta representan a adultos mayores de 55 a 64 años y por último los de color café representan a los ancianos de más de 65 años. La Figura 6 muestra la representación de los agentes poblando el medio ambiente virtual.

La implementación de las ecuaciones de velocidad propuestas por Samson permiten observar desplazamientos de tipo natural. Además de la creación de los LCP, los agentes tienen la posibilidad de tomar diferentes caminos ocupando todo

el espacio geográfico del ambiente permitiendo lograr comportamientos parecidos a los reales.

En la Figura 6 se muestra la simulación de los agentes desplazándose por las calles de la ciudad, rumbo a su LCP objetivo, se observa como los agentes toman en cuenta al resto de los agentes dentro del ambiente evitando colisionar con otros. Se puede ver así mismo como se comienza acumular una masa importante de agentes intentado acceder al recurso, lo que produce saturación en determinadas zonas. Los LCP permiten concentrar agentes de un mismo tipo, sin embargo es válido encontrar agentes de otro tipo en la misma zona, esto es debido a las rutas que toman los agentes y son comportamientos similares a los vistos en la realidad. Finalmente la emergencia de comportamientos como la agrupación de multitudes resulta de la necesidad de acceder a un recurso, tal como sucede en la realidad.

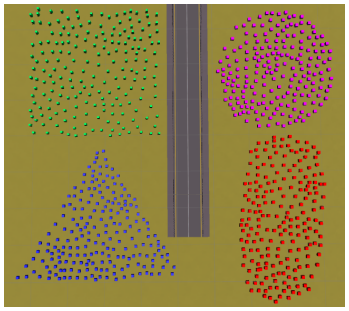

Fig. 7. Patrones de multitudes obtenidas por los agentes.

Uno de los aspectos importantes de las multitudes es la forma en cómo estas se agrupan [18]. Los expertos en comportamiento humano han determinado ciertos patrones en los que destacan la agrupación cuadrada, circular, delta y ovoide. El siguiente resultado se obtuvo al forzar a los agentes a agruparse en estos patrones, sin que exista una orden directa, esto lo hacen de acuerdo a la negociación de ocupar un espacio, sabiendo que la tarea es completa en el momento en que todos los agentes logren el patrón. La Figura 7 muestra los patrones obtenidos por los agentes, es importante resaltar que la distancia entre agentes no está dada, en caso contrario el resultado final sería una formación tipo militar (alineada), lo que no sucede en el comportamiento de una población real de civiles.

En muchas de las manifestaciones que involucran multitudes estas se agrupan en pequeños grupos y se van uniendo a otros para llegar al objetivo final. Estos pequeños grupos deben sortear diversos obstáculos para alcanzar al contingente mayor, en nuestra siguiente prueba a manera de experimentación lo que hacemos es integrar dos grupos de multitudes con diferente patrón y observar como se realiza la integración en un solo grupo y que patrón se obtiene como resultado. Esté comportamiento se observa en la Figura 8, en el momento en que comienzan el desplazamiento estos rompen el patrón inicial y una vez que se forma el

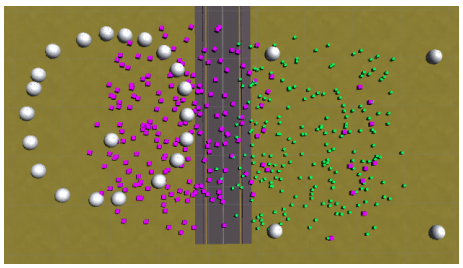

Fig. 8. Patrones de multitudes obtenidas por los agentes.

nuevo grupo se mantiene un patrón de forma cuadrada cuando se trata de desplazamiento y de forma circular u ovoide en el caso de espera. El patrón delta se forma en presencia de un agente líder a seguir.

En el vídeo [1] en la web se pueden observar dos experimentaciones en donde los agentes se desplazan a sus objetivos negociando espacios y calculando sus rutas, también en el mismo vídeo se puede observar la integración de las dos multitudes que se muestran en la Figura 8.

5. Conclusiones

El uso de las velocidades de desplazamiento de los seres humanos utilizado en las simulaciones de multitudes, permite observar comportamientos parecidos a los reales. La categorización de los individuos por edades genera comportamientos indistintos, lo que permite observar en la simulación comportamientos lo más parecidos a la realidad. Una de las aportaciones que se realizan en este trabajo es la creación de LCPs utilizando diagrama de Voronoi lo que nos permite generar concentraciones de individuos en un espacio al cual los agentes necesitan acceder como recurso, además de ver como los agentes realizan procesos de comunicación y negociación para el logro de sus objetivos individuales. La identificación de LCPs permite observar concentración heterogénea de agentes de acuerdo a un contexto, por ejemplo un parque deberá ser poblado en su mayoría por ancianos, niños y mujeres. Poder reproducir los patrones de las agrupaciones de las multitudes con los agentes virtuales nos permite estudiar los comportamientos como suceden en la vida real, permitiendo a las organizaciones evaluar las condiciones e implementar las políticas de posibles contingencias.

References

1. Bleiweiss, A.: Multi agent navigation on the gpu. In: GDC09 Game Developers Conference. vol. 2009 (2009)

[1] `https://youtu.be/GaEpYtvak04`

2. Chenney, S.: Flow tiles. In Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation - SCA pp. 233–242 (2004)
3. De Berg, M., Van Kreveld, M., Overmars, M., Schwarzkopf, O.C.: Computational geometry. In: Computational geometry, pp. 147–169. Springer (2000)
4. De Gyves, O., Toledo, L., Rudomin, I.: Comportamientos en simulación de multitudes: revisión del estado del arte. Research in Computer Science pp. 319–334 (2013)
5. Fortune, S.: A sweepline algorithm for voronoi diagrams. Algorithmica 2(1-4), 153–174 (1987)
6. G, L.: Psychologie des foules. Alcan, Paris (1895)
7. Guy, S.J., Chhugani, J., Curtis, S., Dubey, P., Lin, M., Manocha, D.: Pledestrians: a least-effort approach to crowd simulation. In: Proceedings of the 2010 ACM SIGGRAPH/Eurographics symposium on computer animation. pp. 119–128. Eurographics Association (2010)
8. Hadap, S., Eberle, D., Volino, P., Lin, M.C., Redon, S., Ericson, C.: Collision detection and proximity queries. In: ACM SIGGRAPH 2004 Course Notes. p. 15. ACM (2004)
9. Helbing, D., Molnar, P.: Social force model for pedestrian dynamics. Physical review E 51(5), 4282 (1995)
10. Kapadia, M., Singh, S., Reinman, G., Faloutsos, P.: A behavior-authoring framework for multiactor simulations. Computer Graphics and Applications, IEEE 31(6), 45–55 (2011)
11. Millan, E., Hernández, B., Rudomin, I.: Large crowds of autonomous animated characters using fragment shaders and level of detail. ShaderX5: Advanced Rendering Techniques pp. 501–510 (2007)
12. Paolo Fiorini, Z.S.: Motion planning in dynamic environments using velocity obstacles. The International Journal of Robotics Research pp. 760–772 (July 1998)
13. Pelechano, N., Allbeck, J.M., Badler, N.I.: Controlling individual agents in high-density crowd simulation. In: Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation. pp. 99–108. Eurographics Association (2007)
14. Pettré, J., Ciechomski, P.d.H., Maïm, J., Yersin, B., Laumond, J.P., Thalmann, D.: Real-time navigating crowds: scalable simulation and rendering. Computer Animation and Virtual Worlds 17(3-4), 445–455 (2006)
15. Reynolds, C.W.: Flocks, herds and schools: A distributed behavioral model. ACM SIGGRAPH Computer Graphics (August 1987)
16. Ritterfeld, U., Cody, M., Vorderer, P.: Serious games: Mechanisms and effects. Routledge (2009)
17. Ruiz, S., Hernández, B.: Procesos de decisión de markov y microescenarios para navegación y evasión de colisiones para multitudes. ResearchGate (2014)
18. S. Raupp Musse, D.: A behavioral model for real time simulation of virtual human crowds. IEEE Transactions on Visualization and Computer Graphics 7(2), 152–164 (2001)
19. Samson, M., Crowe, A., De Vreede, P., Dessens, J., Duursma, S., Verhaar, H.: Differences in gait parameters at a preferred walking speed in healthy subjects due to age, height and body weight. Aging Clinical and Experimental Research 13(1), 16–21 (2001)
20. Sun, L., Qin, W.: Simulation of crowd behaviors based on event reaction. In: Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference on. vol. 2, pp. 163–167. IEEE (2011)

21. Thalmann, D.: Populating virtual environments with crowds. In: Proceedings of the 2006 ACM international conference on Virtual reality continuum and its applications. ACM (2006)
22. Thalmann, D., Grillon, H., Maim, J., Yersin, B.: Challenges in crowd simulation. In: 2009 International Conference on CyberWorlds. pp. 1–12. IEEE (2009)
23. Van Den Berg, J., Guy, S.J., Lin, M., Manocha, D.: Reciprocal n-body collision avoidance. In: Robotics research, pp. 3–19. Springer (2011)
24. Zhang, S., Li, M., Li, F., Liu, A., Cai, D.: A simulation model of pedestrian flow based on geographical cellular automata. In: Geoinformatics, 2011 19th International Conference on. pp. 1–5. IEEE (2011)
25. Zhong, Z., Ding, N., Wu, X., Xu, Y.: Crowd surveillance using markov random fields. In: Automation and Logistics, 2008. ICAL 2008. IEEE International Conference on. pp. 1822–1828. IEEE (2008)

ISSN 1870-4069

Creación y clasificación de un corpus criminológico en español usando características lingüísticas superficiales

Luis Gil Moreno[1], Noé Alejandro Castro[1], Juan-Manuel Torres-Moreno[2,3], Luis-Adrián Cabrera-Diego[2], Carlos-Emiliano González-Gallardo[2], Alberto Iturbe[1], Kenia Nieto[1], Arturo-Michel Gómez[1]

[1] Centro Nacional de Investigación y Desarrollo Tecnológico CENIDET, Cuernavaca, México

[2] LIA/Université d'Avignon et des Pays de Vaucluse, Avignon, France

[3] École Polytechnique de Montréal, Montréal, Canada

{luismoreno,ncastro,arturog,iturbe,kenianieto}@cenidet.edu.mx,
{juan-manuel.torres,luis-adrian.cabrera-diego}@univ-avignon.fr,
carlos.gonzalez-gallardo@alumni.univ-avignon.fr

Resumen. Este artículo propone la creación y la caracterización de un corpus especializado en criminología. El corpus está constituido por noticias en texto plano divididas en cinco clases de delitos: homicidio, asalto, secuestro, abuso sexual y extorsión. El objetivo es doble: El primero es crear y anotar manualmente el corpus. Mientras que el segundo objetivo consiste en establecer una clasificación de base usando características lingüísticas superficiales, como los sintagmas nominales y verbales. Los clasificadores utilizados son una Máquina de Soporte Vectorial (SVM) clásico y un modelo Bayesiano.

Palabras clave: Corpus, criminología, clasificación automática, sintagma, máquina de aprendizaje, extracción de información.

Creation and Classification of a Spanish Criminological Corpus using Superficial Linguistic Features

Abstract. This article proposes the creation and characterization of a specialized corpus in criminology. The corpus is composed of news in plain text divided into five classes of crimes: homicide, assault, kidnapping, sexual abuse and extortion. The corpus' objective is twofold: one to create and annotate manually the corpus. Two, to establish a basic classification using superficial linguistic features, such as noun and verb syntagms. We have used two classifiers: a classical Support Vector

Machine (SVM) and Bayesian model.

Keywords. Corpus, criminology, automatic classification, syntagm, machine learning, information extraction.

1. Introducción

De acuerdo con la investigación realizada por [9], México se encuentra entre los 20 países más violentos y peligrosos del mundo. A nivel Centroamérica y el Caribe, México ocupa el segundo lugar en esta clasificación. Por ello es frecuente encontrar en periódicos de circulación nacional, estatal y local, noticias que hagan referencia a diferentes delitos, como el secuestro, el asalto, el abuso sexual, entre otros.

En este artículo se proponen dos tareas relacionadas con el tema evocado. La primera de ellas es la creación de un corpus especializado de noticias delictivas usando diferentes periódicos del Estado de Morelos, México. La segunda tarea es la utilización de herramientas de Procesamiento del Lenguaje Natural (PLN) con el objetivo de clasificar automáticamente noticias que traten sobre delitos. Específicamente, se busca crear un clasificador de noticias que permita determinar cuáles son los delitos tratados al interior de una nota informativa, para que de esta forma se establezca un *baseline*. Los delitos a identificar por el clasificador son aquellos denominados de *Alto impacto* [15].

El artículo está organizado de la siguiente manera: en la Sección 2. se presenta el estado del arte. En la Sección 3. se encuentra la descripción del corpus textual de delitos y su estadística descriptiva. Posteriormente, en la Sección 4. se da a conocer la metodología que se siguió. En la Sección 5. se presentan los resultados obtenidos de los experimentos. Finalmente, en la Sección 6. se presentan las conclusiones y perspectivas.

2. Estado del arte

En los últimos años se han desarrollado diferentes investigaciones relacionadas al uso del PLN, y especialmente de la Extracción de Información (EI) en temáticas delictivas [12].

Sin embargo, la mayoría de las investigaciones en este campo se han realizado para el idioma inglés. Por ejemplo, [23] utiliza un modelo predictivo con un análisis semántico para inferir posibles crímenes a partir de *tweets*[1]. Otro ejemplo, es el de [10], el cual utiliza técnicas de EI y modelos cognitivos para aumentar la información obtenida de entrevistas con testigos criminales. [14] utiliza métodos de agrupamiento automático *(clustering)*, basados en *k-means*, para encontrar patrones criminales. [11] utiliza medidas de similitud y de aprendizaje automático para analizar y clasificar textos que describen crímenes, ya

[1] `http://www.twitter.com`

sean distintos, similares o los mismos. [6] utiliza una red de neuronas artificiales (*Artificial Neural Network, ANN)* para encontrar patrones de clasificación en bases de datos delictivas de la policía. Finalmente, en el trabajo desarrollado por [4], se procesan los reportes delictivos de la policía de Arizona en EE.UU, en búsqueda de elementos relevantes, por ejemplo, nombres de personas, drogas, armas o eventos criminales.

Para el idioma portugués, se encuentra el trabajo de [17] en el que los autores emplean la EI y un análisis semántico para enriquecer automáticamente la información sobre crímenes presentes en el sitio colaborativo *WikiCrimes*[2]. Por otro lado, para el idioma alemán se encontró el trabajo [22] en el cual se propone la creación de un corpus especializado del plagio con la intención de utilizarlo para la detección de plagio de documentos. No obstante, para el idioma español, según nuestro conocimiento, no existen investigaciones similares.

La importancia de la clasificación automática de textos criminales, recae en el hecho de que puede ser empleado en un análisis criminalístico [7] y para la detección de entidades criminales [5]. Incluso, la clasificación de noticias puede ayudar a encontrar patrones en reportes delictivos u otro tipo de aspectos criminalísticos [14].

2.1. Características utilizadas

Con base en la investigación presentada por [12], se puede concluir que la mayoría de las trabajos que buscan patrones criminales usan elementos específicos como armas de fuego o armas blancas. Sin embargo, existen otras investigaciones que emplean el PLN, más específicamente la EI, para llevar a cabo esta tarea.

En el trabajo realizado por [12] se realiza un estudio exhaustivo sobre los diferentes entornos de trabajo o *Frames* existentes, los cuales analizan patrones en búsqueda de tendencias criminales. Aunque la mayoría utiliza sensores en búsqueda de elementos más específicos como armas de fuego o armas blancas, existen algunos que trabajan directamente con herramientas de PLN.

Las técnicas de EI son variadas y muchas veces se realizan con base en la frecuencia de términos en los textos. En [13], se utiliza el modelo TF-IDF donde se vectorizan los textos de entrada; los términos con una frecuencia elevada así como las palabras vacías se discriminan, mientras que aquellos con frecuencia única, se consideran como Entidades Nombradas (EN).

En el trabajo de [18] se ocupa la misma técnica, sólo que esta vez el modelo TF-IDF se emplea para la extracción de características. Posteriormente, las características extraídas se utilizan para clasificar los textos usados ocupando la medida de similitud coseno.

Analizando los resultados de los trabajos mencionados, se optó por implementar dichas técnicas para la extracción de características, con la diferencia de proponer un modelo diseñado para textos cortos. Esto para permitir mantener la misma calidad en las características extraídas, pero aumentando la velocidad

[2] `http://www.wikicrimes.org`

de procesamiento. Se observó que se podría realizar una adaptación de los trabajos de [1] y [19]. En el primero de ellos, se hace una adaptación del modelo TF-IDF para textos cortos y en el segundo se toman bigramas poco comunes como candidatas a EN. Con estas técnicas se podrán realizar búsquedas de términos relevantes en las noticias y considerarlas entonces como características que pudieran describir cada clase que se tiene como objeto de estudio en esta investigación.

3. Corpus textual y estadística descriptiva

A continuación, se describe el proceso que se siguió para la conformación del corpus presentado en este trabajo. De igual manera se detalla la información estadística sobre el mismo.

3.1. Construcción del corpus

Para la conformación del corpus, fue necesario la descarga de noticias a través de diversos portales periodísticos locales en el Estado de Morelos, México. Se eligieron periódicos de circulación local, ya que son en estos medios en donde se reporta mayormente la actividad delictiva de la zona.

Para la extracción de noticias, se desarrolló un módulo ocupando la biblioteca JSoup [8]. Esta analiza la estructura HTML de una página web para ubicar y extraer específicamente la información deseada. En este caso, lo que interesa para el estudio es el cuerpo de la nota periodística.

El corpus fue constituido con noticias descargadas de los siguientes periódicos:

- La Unión de Morelos[3],
- El Diario de Morelos[4],
- La Jornada de Morelos[5].

Se descargaron en total 1 000 noticias que fueron almacenadas en texto plano en formato *utf8*. Los documentos recuperados cumplen la condición de reportar al menos uno de los siguientes delitos:

1. Homicidio,
2. Asalto,
3. Secuestro,
4. Abuso sexual.

Cabe destacar que al momento de etiquetar las noticias, los anotadores se percataron de la existencia de una clase adicional presente en un número

[3] http://www.launion.com.mx
[4] http://www.diariodemorelos.com
[5] http://www.jornadamorelos.com

significativo de notas periodísticas: la extorsión. Por tanto, se agregó esta clase como uno de los posibles delitos del corpus.

Las noticias fueron descargadas en dos períodos de tiempo, para evitar una tendencia sobre ciertos hechos específicos. Por ejemplo, la descarga de todas las noticias posteriores al asesinato de un presidente, implicaría que la mayoría de las notas descargadas serían destinadas a la clase homicidio. Así, el primer intervalo de tiempo de descarga de noticias fue del 11 al 15 de abril de 2016, y el segundo intervalo de descargas tuvo lugar del 07 al 09 de septiembre de 2016.

3.2. Proceso de anotación manual

Para la tarea de anotación manual del corpus de noticias, se seleccionaron cuatro estudiantes universitarios con nivel de maestría. El corpus completo anotado de noticias se nombró como Corpus Anotado de Delitos (CAD)[6].

A cada una de estas personas se les proporcionaron aleatoriamente y sin repetición 250 noticias. Considerando que cada noticia posee en promedio 371.6 palabras, significa que cada uno de ellos tuvo que analizar en total un promedio de 93 000 palabras.

Cada noticia fue clasificada manualmente en al menos una de las cinco posibles clases según la información contenida. En otras palabras, una noticia puede contener múltiples actos delictivos y por consiguiente, pertenecer a más de una clase. Esto fue detectado durante el proceso de etiquetado manual.

Finalmente, las cinco clases retenidas para el presente experimento fueron las siguientes:

1. Homicidio,
2. Asalto,
3. Secuestro,
4. Abuso sexual,
5. Extorsión.

Se midió el tiempo usado por los anotadores en el experimento. La tarea de anotación se llevó a cabo en 17 horas. La anotación de cada noticia necesitó de aproximadamente tres minutos.

En la Tabla 1 se muestran las estadísticas básicas del corpus CAD.

3.3. Palabras clave de las clases

Además de la anotación manual hecha por los cuatro anotadores, se les pidió que estos realizaran una lista de las palabras claves que les permitiera clasificar las noticias. De esta forma, no solamente se obtuvo una anotación manual del corpus, sino también un conjunto de términos, mono o multipalabra, que se usan de manera recurrente en el corpus. En la Tabla 2 se muestran los términos, en su forma canónica, encontrados por los anotadores para cada clase.

[6] El corpus CAD podrá ser solicitado a través del correo: `luismoreno@cenidet.edu.mx`

Tabla 1. Corpus CAD en función de los documentos.

Clases	La Unión de Morelos	El Diario de Morelos	La Jornada de Morelos	Total	Palabras (tokens)
Homicidio	139	130	125	394	146 410
Asalto	145	160	136	441	164 990
Secuestro	101	95	109	305	113 338
Abuso Sexual	45	48	62	155	55 598
Extorsión	36	69	45	150	55 740

Tabla 2. Palabras clave del corpus CAD.

Clase	Palabras clave
Asalto	robo, sustraer, hurtar, amenazar, despojar, interceptar, quitar, desvalijar, desmantelar, sorprender en posesión, ocultar
Homicidio	encontrar sin vida, linchamiento, asesinar, atropellar, hallar muerto, cadáver, disparar, baleado, balazo, acribillar, atacar a tiro, persona sin vida, morir, dar disparos
Secuestro	secuestro, subir a la fuerza, forcejear, libertad, levantar persona, víctima rescatada, privación, liberar víctima, llevar a la fuerza, rescate
Abuso sexual	violación, agredir de forma sexual, íntima, estupro
Extorsión	extorsionar, golpear, obligar, intimidar, amenazar

Como se puede observar, la gran mayoría de las palabras claves encontradas en cada clase corresponden a verbos (por ejemplo: *despojar, forcejear, disparar*) y sustantivos (*privación, estupro, cadáver*). Sin embargo, también se encuentran algunas formaciones como verbo-sustantivo (*levantar-persona, liberar-víctima*), verbo-adjetivo (*hallar-muerto*) y adjetivo (*íntima, baleado*).

4. Metodología

La metodología propuesta en este artículo esté dividida en dos partes: Extracción de las características (Sección 4.1.) y Clasificación de noticias por el contenido (Sección 4.2.).

4.1. Extracción de características

Este proceso se basó en las conclusiones y resultados obtenidos por [3]. Este artículo argumenta que son los sintagmas nominales los que mejor describen la información de un texto. En este caso en particular, interesa la identificación del tipo de delito que se reporta en la nota. Para este proyecto consideramos que, los actos que se detallan en el texto, pueden analizarse mediante la identificación de los verbos. Lo anterior, coincide con las conclusiones de los anotadores, quienes observaron que los sustantivos y verbos son los que más información proveen

para la detección del tipo de delito. Por tanto, también se estudiarán sintagmas verbales.

La adecuada extracción de los sintagmas es de vital importancia para las actividades posteriores, ya que esta será la que defina el conjunto de características que constituirán una bolsa de palabras. La bolsa de palabras será la representación utilizada tanto en la fases de aprendizaje como en la fase de pruebas de los clasificadores.

El proceso que se sigue para la extracción de las características se describe a continuación:

Primeramente, cada uno de los textos se anota con etiquetas que indican la categoría gramatical de las palabras (POS o *Part-of-Speech* en inglés), usando la herramienta Freeling [16].

Después, a partir de las etiquetas POS de las palabras de cada texto, se extraen los siguientes patrones sintácticos:

- Sintagmas verbales (VP)[7],
- Sustantivos,
- Verbos.

Una vez extraídos estos patrones sintácticos, se calcula el grado de importancia de cada palabra que aparece en los sintagmas. Esto se realiza con base en las investigaciones de [2]. Para llevar a cabo esto, se multiplica el número de palabras que compone el sintagma (|VP|) por la frecuencia de cada palabra del sintagma (UF o Unigram Frequency en inglés), dicha operación se formaliza en la Ecuación 1:

$$\mathrm{UF(VP)} = \sum_{i=0}^{|\mathrm{VP}|} \text{Unigram Frequency}(w_i). \tag{1}$$

Posteriormente, según con el modelo propuesto por [2], el resultado obtenido en la Ecuación 1 se multiplica por la frecuencia del sintagma en el artículo, (VPF(VP)). El resultado se divide entre la cantidad de palabras que compone el sintagma (|VP|):

$$Score(\mathrm{VP}) = \frac{\mathrm{UF(VP)} \cdot \mathrm{VPF(VP)}}{|\mathrm{VP}|}. \tag{2}$$

Una vez que se han determinado los grupos y sus elementos, se calcula su puntuación. El cálculo de lo antes mencionado no es más que una media aritmética. Siendo más específicos, esto se hace mediante la sumatoria del *score* de cada elemento, perteneciente al grupo, sobre la cantidad de elementos que pertenecen al mismo grupo. En la Ecuación 3 se presenta la fórmula utilizada para el cálculo del *score* de cada grupo:

$$Score(Grupo) = \frac{\sum_{i=0}^{|Grupo|} Score(\mathrm{VP}_i)}{|Grupo|}. \tag{3}$$

[7] Son aquellas construcciones que se componen de un verbo y su complemento.

Finalmente, se estableció un umbral para delimitar las palabras o sintagmas más importantes del texto procesado. Estas palabras son las que conformarán las bolsas de palabras o características que definirán cada clase de delitos. El umbral se estableció con respecto a los valores medios de cada clase, eliminando así aquellos *scores* de grupos demasiado elevados. Tal es el caso de *stopwords* o palabras que no dan ninguna descripción relevante sobre el documento. Igualmente, representan aquellas clases con *scores* muy bajos, como lo son los nombres de personas, lugares y fechas (las Entidades Nombradas).

4.2. Clasificación *baseline* del corpus

Con el propósito de establecer una medida básica de desempeño, se decidió utilizar dos clasificadores sobre el corpus anotado. El corpus CAD fue dividido en un corpus de aprendizaje (CA) y un corpus de prueba (CP). La distribución de noticias en cada subcorpus fue obtenida aleatoriamente con una distribución uniforme. Esto se llevó a cabo, para garantizar la misma distribución que en el corpus CAD. La tarea consistió entonces en determinar a qué clase pertenece cada noticia.

El corpus de aprendizaje CA está formado por un subconjunto de noticias del 70% del total del corpus CAD, y el corpus de prueba del 30% restante. Estos subconjuntos de noticias se sometieron al proceso de extracción de características descrito en la Sección 4.1.. Como datos de entrenamiento, fueron ocupadas las características extraídas del conjunto de noticias anotadas manualmente (ver Sección 3.2.). Estas características fueron usadas para analizar la clase a la que corresponden las noticias del corpus CP.

Para la clasificación del CP, se empleó la plataforma Weka[8], que permite trabajar con diversos algoritmos de clasificación. En este estudio se realizaron pruebas con un modelo Bayesiano (Naïve Bayes) y con una Máquina de Soporte Vectorial (SVM).

5. Resultados y evaluación

Los resultados que se presentan a continuación, contemplan tres experimentos. En el primero se sometió a análisis la noticia completa: Tabla 3 y Tabla 4. En el segundo, fueron utilizados únicamente el título de la noticia y el primer párrafo: Tabla 5 y Tabla 6. Finalmente, se hizo la última prueba considerando únicamente el título de la noticia: Tabla 7 y Tabla 8. En todos los casos, para la evaluación se utilizó la medida clásica F-Score, definida por la Ecuación 4:

$$\text{F-Score} = \frac{2 \times (\text{Precisión} \times \text{Recall})}{\text{Precisión} + \text{Recall}}. \tag{4}$$

En la columna "Media" de las Tablas 3–8, se indica el promedio de Precisión, *Recall* y *F-Score* de cada experimento.

[8] `http://www.cs.waikato.ac.nz/ml/weka`

Tabla 3. Resultados de la clasificación (Noticia completa - SVM).

Clases	Asalto	Homicidio	Secuestro	Abuso Sexual	Extorsión	Media
Precisión	0.7348	0.7613	0.6851	0.6000	0.8032	0.7169
Recall	0.8818	0.6146	0.6851	0.3846	0.9074	0.6947
F-Score	0.8016	0.6802	0.6851	0.4687	0.8521	0.6975

Tabla 4. Resultados de la clasificación (Noticia completa - Naïve Bayes).

Clases	Asalto	Homicidio	Secuestro	Abuso Sexual	Extorsión	Media
Precisión	0.7638	0.8033	0.6143	0.5625	0.7123	0.6912
Recall	0.8818	0.4495	0.7963	0.4615	0.9630	0.7104
F-Score	0.8186	0.5765	0.6935	0.5070	0.8189	0.6829

Tabla 5. Resultados de la clasificación (Título de la noticia y primer párrafo - SVM).

Clases	Asalto	Homicidio	Secuestro	Abuso Sexual	Extorsión	Media
Precisión	0.9167	0.6320	0.5200	0.5161	0.8636	0.6897
Recall	0.7000	0.7248	0.4815	0.4103	0.7037	0.6040
F-Score	0.7938	0.6752	0.5000	0.4571	0.7755	0.6403

Tabla 6. Resultados de la clasificación (Título de la noticia y primer párrafo - Naïve-Bayes).

Clases	Asalto	Homicidio	Secuestro	Abuso Sexual	Extorsión	Media
Precisión	0.9310	0.6737	0.5192	0.5278	0.6719	0.6647
Recall	0.7364	0.5872	0.7297	0.4872	1.1316	0.7344
F-Score	0.8223	0.6275	0.6067	0.5067	0.8431	0.6813

Tabla 7. Resultados de la clasificación (Título de la noticia - SVM).

Clases	Asalto	Homicidio	Secuestro	Abuso Sexual	Extorsión	Media
Precisión	0.8971	0.4167	0.5833	0.7200	0.8000	0.6834
Recall	0.5545	0.8716	0.6034	0.1915	0.6207	0.5683
F-Score	0.6854	0.5638	0.5932	0.3025	0.6990	0.5688

Se puede observar, que no existe una gran diferencia entre los resultados dados por los SVM ni por los modelos Bayesianos. Esto se puede deber a que los SVM no fueron optimizados sobre el corpus CAD. A pesar de ello, la Media de F-Score usando SVM es la más elevada, con F-Score = 0.6975. Cambiando los parámetros del SVM se podrían obtener todavía mejores resultados.

A partir de lo expresado por los anotadores, quienes en diversas ocasiones

Tabla 8. Resultados de la clasificación (Título de la noticia - Naïve Bayes).

Clases	Asalto	Homicidio	Secuestro	Abuso Sexual	Extorsión	Media
Precisión	0.9405	0.4316	0.6271	0.7273	0.6897	0.6832
Recall	0.7182	0.7523	0.7400	0.2553	0.8000	0.6532
F-Score	0.8144	0.5485	0.6789	0.3780	0.7407	0.6321

encontraron más de un delito en una misma noticia, se considera que una forma de mejorar los resultados (en Precisión, Recall y F-Score) podría consistir en analizar las noticias con un clasificador multiclase. Un clasificador de este tipo permitiría incluir una misma noticia en dos o más clases. Incluso, se podría mejorar el F-Score en noticias ligadas a "Homicidio", "Secuestro" y "Abuso sexual" que presentan los porcentajes más bajos, debido al gran recubrimiento que existe entre estas.

6. Conclusiones

En este artículo se ha introducido el Corpus Anotado de Delitos en México, CAD. Se ha caracterizado este corpus y se han presentado algunas medidas *baseline* para la clasificación automática de cinco categorías de delitos. El corpus CAD puede ser utilizado en tareas de clasificación de delitos usando herramientas de PLN. Estas herramientas de análisis podrían ser de utilidad para las diferentes instancias de gobierno (policía, institutos de la juventud, etc.) así como para organizaciones descentralizadas (comisiones de derechos humanos) o no gubernamentales.

Algunos ejemplos de posibles herramientas, son los mapas donde se indican los delitos de alto impacto, los buscadores de noticias delictivas, las herramientas de documentación y la generación de síntesis de delitos [20] entre otros.

A futuro, se estudiará si este esquema de clasificadores clásicos u otros, incluyendo Redes de Neuronas Artificiales (ANN) de tipo incremental [21], podrían funcionar con textos cortos (resúmenes de noticias, *tweets*, denuncias por Internet, etc.). Igualmente, se planea aumentar el número de noticias del corpus CAD, para abarcar diferentes periodos de tiempo.

Agradecimientos. Este trabajo fue financiado parcialmente por el *Programa de Becas Mixtas de CONACYT* (México), el *Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET)* (México) y por el *Laboratoire Informatique d'Avignon (LIA)* de la *Université d'Avignon et des Pays de Vaucluse* (Francia).

Referencias

1. Bollen, J., Mao, H., Pepe, A.: Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. ICWSM 11, 450–453 (2011)

2. Bracewell, D.B., Ren, F., Kuriowa, S.: Multilingual single document keyword extraction for information retrieval. In: 2005 International Conference on Natural Language Processing and Knowledge Engineering. pp. 517–522. IEEE (2005)
3. Bracewell, D.B., Yan, J., Ren, F., Kuroiwa, S.: Category classification and topic discovery of japanese and english news articles. Electronic Notes in Theoretical Computer Science 225, 51–65 (2009)
4. Chau, M., Xu, J.J., Chen, H.: Extracting meaningful entities from police narrative reports. In: Proceedings of the 2002 annual national conference on Digital government research. pp. 1–5. Digital Government Society of North America (2002)
5. Chen, H., Chung, W., Xu, J.J., Wang, G., Qin, Y., Chau, M.: Crime data mining: a general framework and some examples. Computer 37(4), 50–56 (2004)
6. Dahbur, K., Muscarello, T.: Classification system for serial criminal patterns. Artificial Intelligence and Law 11(4), 251–269 (2003)
7. Estivill-Castro, V., Lee, I.: Data mining techniques for autonomous exploration of large volumes of geo-referenced crime data. In: Proc. of the 6th International Conference on Geocomputation. pp. 24–26. Citeseer (2001)
8. Hedley, J.: Java html parser. `http://jsoup.org/` (2016)
9. Institute for Economics & Peace: Índice de Paz en México 2015, Un análisis de la dinámica de los niveles de paz en México. Report IEP 31, México (2015)
10. Ku, C.H., Iriberri, A., Leroy, G.: Crime information extraction from police and witness narrative reports. In: Technologies for Homeland Security, 2008 IEEE Conference on. pp. 193–198. IEEE (2008)
11. Ku, C.H., Leroy, G.: A decision support system: Automated crime report analysis and classification for e-government. Government Information Quarterly 31(4), 534–544 (2014)
12. Kumar, A.S., Gopal, R.K.: Data mining based crime investigation systems: Taxonomy and relevance. In: Communication Technologies (GCCT), 2015 Global Conference on. pp. 850–853. IEEE (2015)
13. Lee, S., Kim, H.j.: News keyword extraction for topic tracking. In: Networked Computing and Advanced Information Management, 2008. NCM'08. Fourth International Conference on. vol. 2, pp. 554–559. IEEE (2008)
14. Nath, S.V.: Crime data mining. In: Advances and Innovations in Systems, Computing Sciences and Software Engineering, pp. 405–409. Springer (2007)
15. Observatorio Nacional Ciudadano Seguridad, Justicia y Legalidad: Reporte sobre delitos de alto impacto Junio 2016. Reporte Año 3, No. 5, México (2016)
16. Padró, L., Stanilovsky, E.: Freeling 3.0: Towards wider multilinguality. In: Proceedings of the Language Resources and Evaluation Conference (LREC 2012). ELRA, Istanbul, Turkey (May 2012)
17. Pinheiro, V., Furtado, V., Pequeno, T., Nogueira, D.: Natural language processing based on semantic inferentialism for extracting crime information from text. In: Intelligence and Security Informatics (ISI), 2010 IEEE International Conference on. pp. 19–24. IEEE (2010)
18. Quinteiro-González, J.M., Martel-Jordán, E., Hernández-Morera, P., Ligero-Fleitas, J.A., López-Rodriguez, A.: Clasificación de textos en lenguaje natural usando la wikipedia. Iberian Journal of Information Systems and Technologies (8), 39–52 (2011)
19. Shinyama, Y., Sekine, S.: Named entity discovery using comparable news articles. In: Proceedings of the 20th international conference on Computational Linguistics. p. 848. Association for Computational Linguistics (2004)
20. Torres-Moreno, J.M.: Automatic Text Summarization. Wiley and Sons (2014)

21. Torres-Moreno, J.M., Gordon, M.: Efficient adaptive learning for classification tasks with binary units. Neural Computation 10(4), 1007–1030 (1998)
22. Torres-Moreno, J.M., Sierra, G., Peinl, P.: A German Corpus for Similarity Detection Tasks. International Journal of Computational Linguistics and Applications 2(5), 9–22 (2014)
23. Wang, X., Gerber, M.S., Brown, D.E.: Automatic crime prediction using events extracted from twitter posts. In: International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction. pp. 231–238. Springer (2012)

ISSN 1870-4069

Análisis comparativo entre diferentes entornos de aprendizaje automático para el análisis de sentimientos

Karen L. Vazquez[1], Mireya Tovar[1], José A. Reyes-Ortiz[2], Darnes Vilariño[1]

[1]Benemérita Universidad Autónoma de Puebla,
Facultad de Ciencias de la Computación, Puebla, México

[2]Universidad Autónoma Metropolitana,
Departamento de Sistemas, Azcapotzalco, México

krnlet@gmail.com,{mtovar,darnes}@cs.buap.mx, jaro@correo.azc.uam.mx
http://www.lke.buap.mx/

Resumen. El objetivo en este artículo es el estudio del Análisis de Sentimientos con aprendizaje automático y precisamente realizar una comparativa de clasificadores automáticos implementados en la plataforma de software para el aprendizaje automático, WEKA y el lenguaje de programación Python. Los experimentos se elaboraron con tres conjuntos de datos provenientes de SemEval 2016 para la resolución de la tarea 5, subtarea 2 [9]. El enfoque utilizado para el análisis de sentimiento está dividido en 5 fases y los resultados muestran que Python obtiene un mejor desempeño con los clasificadores utilizados.

Palabras clave: Análisis de sentimientos, procesamiento de lenguaje natural, aprendizaje autómatico.

Comparative Analysis of different Machine Learning Environments for Sentiment Analysis

Abstract. The main aim of this paper is the study of Sentiment Analysis with machine learning techniques and to perform a comparison of automatic classifiers implemented in WEKA and Python programming language. The experiments were carried out with three data sets from SemEval 2016 for solving the task 5, subtask 2 [9]. The approach proposed for sentiment analysis is divided into five phases and the results show that Python achieves a better performance than WEKA with the classifiers used.

Keywords. Sentiment analysis, natural language processing, machine learning.

1. Introducción

El Análisis de Sentimientos es una rama del Procesamiento de Lenguaje Natural en el cual se identifica una emoción a través de una oración, frase o expresión escrita en Internet, permitiendo el monitoreo de opiniones sobre diferentes temas discutidos en la Web. En el caso del estudio abordado en este artículo, se analizaron frases u oraciones escritas en español e inglés en las que se opina sobre el servicio de Restaurantes y opiniones escritas en el idioma inglés sobre Laptops. Se realizaron experimentos utilizando 3 clasificadores automáticos: Máquina de Soporte Vectorial (SVM), Naïve Bayes y Naïve Bayes Multinomial, cada uno se prueba con los tres conjunto de datos en el software de aprendizaje automático Weka y en Python, con el fin de realizar una comparación de resultados entre estas dos herramientas.

El artículo se encuentra distribuido de la siguiente manera: en la Sección 2. se presentan algunos trabajos relacionados con el análisis de sentimientos; en la Sección 3. se presenta la parte teórica que sustenta esta investigación; en la Sección 4. se presenta las fases que integran al enfoque propuesto como solución al análisis de sentimientos; en la Sección 5. se muestran los resultados experimentales obtenidos; en la Sección 6. se presenta una comparación entre Python y Weka de acuerdo a los resultados obtenidos en los experimentos y finalmente en la Sección 7. se presentan las conclusiones de esta investigación.

2. Trabajos relacionados

Existen diferentes métodos para el estudio de Análisis de Sentimientos, sin embargo, lo más estudiado es el Aprendizaje Automático y en la literatura existen trabajados que usan diferentes formas de utilizar dicho aprendizaje. A continuación se muestran algunas participaciones en la investigación de la Minería de Opiniones.

Para la solución de una de las tareas de SemEval 2016, en [14] proponen un sistema de aprendizaje para la clasificación de tweets en dos puntos escala, con una arquitectura que consiste en una red neuronal multicapa, dicha red es alimentada de tweets pre procesados como entrada y está predice las etiquetas binarias de los tweets. Los tweets son pre procesados: todos los URLs son codificados dentro de un token URL, las cuentas de usuarios mencionadas en un tweets son codificadas como un token USER, antes de alimentar a la red el tweet se convierte en minúsculas, no se remueven algunas palabras vacías ya que definen relaciones útiles entre palabras y frases, seguido del pre procesado continua la capa de incrustación de los mapas de una secuencia de palabras presentes en el tweet de entrada a la longitud fija correspondiente del valor real de los vectores. En esta investigación utilizan las redes neuronales recurrentes en la arquitectura del sistema, un apilamiento de las capas recurrentes en la parte superior de uno al otro permite la composición de las representaciones semánticas de palabras y frases con el tiempo. La capa *suma* ayuda a combinar la representación del sentimiento específico de las frases a fin de producir una

representación agregada y la salida de esa capa esta conectada con la capa densa que consiste en unidades lineales rectificadas. La capa densa está conectada con la salida de unidades sigmoides que predice la probabilidad de asignar una etiqueta positiva o negativa para el tweet de entrada. Para los experimentos de la red utilizan 30 como longitud máxima de secuencia de entrada, vocabulario de tamaño 400,000 dimensionalidad (d) de incrustación de palabra es de 100, 3 capas recurrentes y se utilizó una deserción del 50% después de cada capa durante el entrenamiento. El sistema produce una recuperación macro promedio de 0.784, mientras que el mejor sistema obtuvo 0.797 de puntaje.

En el trabajo presentado por [10], se propone una metodología para el análisis de sentimientos multimodales, consiste en sentimientos de vídeos de la Web mediante la demostración de un modelo que utiliza modalidades de audio, visuales y textuales como fuentes de información. La extracción de características visuales se hace por medio de un analizador de expresión facial y clasifica las expresiones faciales con el fin de definir las categorías de sentimientos, se utilizaron polaridades positivas, negativas y neutras como clases de sentimientos. En el experimento se utilizan las características extraídas por FSDK 1.7 junto con las extraídas utilizando GAVAM, usaron ELM para construir el modelo de análisis de sentimientos de expresiones faciales. Se produjó una precisión de 68.60%. Para las características de audio se extrajeron mediante un marco del tipo de 30 Hz y una ventana deslizante de 100 ms y para calcular las características se utilizó el software OpenEar. En el caso de la identificación de los sentimientos en el texto se siguió el paradigma de computación Sentic, el cual considera el texto como expresión de ambos: *semantics y sentics*. Los experimentos se realizaron con el conjunto de datos de Youtube y con diferentes clasificadores supervisados: Naïve Bayes, SVM, ELM, redes neuronales y los mejores resultados se obtuvieron con el clasificador ELM.

En [6] se llevó acabo una metodología para determinar la posible popularidad, opinión y sentimiento de un producto en diferentes localidades a través de usuarios de género masculino y femenino, el análisis es en tweets sobre el iPhone 6. La extracción de estos tweets se realizó por medio de la API de Twitter, estos tweets contienen datos no relevantes es por eso que los autores realizaron una limpieza básica usando Java. Utilizaron una herramienta de Procesamiento de Lenguaje Natural de Stanford y SentiWordNet. Para determinar el género de los usuarios de los tweets utilizaron *NamSor* que es una herramienta de minería de datos y *Rapid Miner* como extensión para la clasificación de género. Los autores realizaron comparaciones con diferentes escenarios del mundo real, y los resultados muestran sentimientos negativos hacia la pantalla y el touch del iPhone 6, y sentimientos positivos hacia la cámara que es bien recibida por los usuarios generales y revisores.

Siguiendo la técnica de aprendizaje automático con datos de twitter, en [13] automáticamente proporcionan la opinión sobre un producto. El pre procesado de datos es la parte más importante en el proceso y comprende el reemplazo de emoticones, gestion de URL, hash-tags, espacios en blanco, identificación de puntuación y conversión de mayúsculas. En la extracción de datos se utilizó

la técnica basada a nivel de oración y consta de Tokenización, POS tagger, Stopwords, SentiWordNet, clasificación y evaluación usando el clasificador SVM, el sistema cálcula el porcentaje de tweets positivos y negativos de un producto en particular.

En el caso del trabajo desarrollado en [7] se hace uso de tweets, como las reseñas escritas por usuarios hacia una película en particular. En el pre procesado de los tweets hacen uso de Procesamiento de Lenguaje Natural (por sus siglas en inglés PLN), este pre procesado se refiere a la limpieza y normalización del texto, para hacer el análisis de sentimientos (stop words, puntuación, palabras duplicadas, caracteres repetidos, emoticones y acrónimos de Internet y, URLs). Eligieron el clasificador Naïve Bayes como técnica aplicada para ciertas clases de problemas. Asignan pesos a varios factores como el número de vistas en cada película, el número de tweets, etc., los pesos de todos los factores son combinados para predecir el éxito total de taquilla sobre la película.

Para el desarrollo de este trabajo, se consideran tres clasificadores de aprendizaje automático supervisado, algunos de ellos son utilizandos en la literatura mencionada anteriormente y muestran buenos resultados en el Análisis de Sentimientos. Máquina de Soporte Vectorial (SVM), Naïve Bayes y Naïve Bayes Multinomial son los estudiados en esta investigación, para cada uno se utiliza como característica léxica a los unigramas y un esquema de pesado como TF-IDF, se emplean tres conjuntos de datos con dos dominios y dos idiomas. Adicionalmente se realiza una comparacíon entre dos tipos de lenguajes de aprendizaje automático como son Python y Weka.

3. Aprendizaje supervisado

Como clasificador de polaridad se utiliza aprendizaje automático supervisado, el cual comprende algoritmos automáticos de clasificación los aplicados son: Máquina de Soporte Vectorial (SVM), Naïve Bayes y Naïve Bayes Multinomial.

3.1. Máquina de soporte vectorial

Máquina de Soporte Vectorial (SVM) es una técnica para la clasificación de datos [2]. El objetivo de las máquinas de soporte vectorial es producir un modelo que se base en los datos de entrenamiento que predice los valores objetivo de los datos de prueba, dado únicamente los atributos de los datos de prueba. Las máquinas de soporte vectorial en Python cuentan con métodos de aprendizaje supervisado utilizados para la clasificación, regresión y detección de valores atípicos. La herramienta **scikit-learn** [3] de Python es una máquina de aprendizaje en Python, que proporciona simples y eficientes herramientas para la minería de datos y análisis de sentimientos, contiene varios *vectorizers*[1] de traducción para los documentos de entrada en vectores de características.

[1] Función utilizada en scikit-learn http://scikit-learn.org/

3.2. Naïve Bayes

El clasificador Bayesiano, asigna la clase más probable dado un ejemplo descrito por su vector de características [12]. Naïve Bayes es un método sencillo e intuitivo con un funcionamiento similar a otros enfoques. Naïve Bayes combina la eficiencia (tiempo óptimo de rendimiento) con una razonable precisión [5]. Este clasificador tiene como inconveniente asumir independencia condicional entre los rasgos lingüísticos. Si las características principales son los tokens extraídos de los textos es evidente que no pueden considerarse como independientes, ya que las palabras co-ocurren en un texto siendo de alguna manera unidas por diferentes tipos de dependencias sintácticas y semánticas. Pero incluso si Naïve Bayes produce un modelo muy simplificado, sus decisiones de clasificación son precisas. Este modelo probabilistico esta basado en la regla de Bayes, junto con un fuerte supuesto de independencia, dicho supuesto es que dada una clase (positiva o negativa) las palabras son condicionalmente independientes entre si, el supuesto no afecta tanto a la exactitud en la clasificación del texto, pero hace más rápido los algoritmos de clasificación aplicables para el problema [8]. Python permite el uso del clasificador Naïve Bayes con el kit de herramientas de lenguaje natural (NLTK)[2].

3.3. Naïve Bayes multinomial

Naïve Bayes Multinomial es un modelo de distribución de palabras en un documento como un polinomio. Un documento se trata como una secuencia de palabras y se supone que cada posición de la palabra se genera independientemente de cualquier otra. Es un clasificador rápido, fácil de implementar y relativamente eficaz [11].El modelo Naïve Bayes Multinomial permite considerar la frecuencia de aparición de cada término en los documentos, esto es importante, ya que podemos suponer que una alta frecuencia de aparición aumenta la probabilidad de pertenecer a una clase particular [1].

4. Enfoque propuesto

Para los objetivos planteados se realizan las siguientes etapas o fases: Pre procesamiento, Extracción de características, Fase de entrenamiento, Fase de prueba y Evaluación. La característica principal es el uso de Frecuencia Inversa del Documento para un Término (TF-IDF), la cual representa el número de documentos en el cual un término dado es calculado y para clasificar las polaridades en las opiniones se utiliza aprendizaje automático supervisado con los clasificadores: Máquina de Soporte Vectorial, Naïve Bayes y Naïve Bayes Multinomial. A continuación se describe a detalle las tres fases propuestas.

– Pre procesamiento:

[2] http://www.nltk.org/book/ch06.html

  - **Colección de opiniones.** Extraer únicamente opiniones de los documentos en formato XML.
  - **Purificación de opiniones.** Proceso en el que se obtienen las opiniones libres de palabras vacías, signos de puntuación, acentos y caracteres aislados.
  - **Tokenización.** Tokenizar cada opinión por palabra.
  - **Stemming.** Proceso en el cual se reducen las palabras de cada opinión, frecuentemente se incluye la eliminación de los afijos derivados.
  - **Filtrado de opiniones.** En esta parte se clasifican las opiniones de entrenamiento por las posibles polaridades: positivo, negativo, neutral y conflicto.
- Extracción de características: Se considera como característica léxica Frecuencia Inversa del Documento para un Término (TF-IDF por sus siglas en inglés) que es una técnica que indica la relevancia de una palabra con respecto al documento seleccionado y al corpus en general, lo cual permite calificar a los documentos del corpus con base a las palabras claves, es decir, si las palabras tiene mayor peso, entonces significa que el documento está más relacionado con las palabras que uno con las mismas palabras pero con menor peso.
- Fase de entrenamiento: Con las características obtenidas en la fase anterior se realiza el proceso de entrenamiento. De acuerdo al algoritmo de clasificación considerado se construye el modelo que posteriormente se utiliza en la fase de prueba. Las clases consideradas son cuatro: positiva, negativa, neutro y conflicto. Los algoritmos de clasificación supervisado usados en esta fase, tanto en WEKA como en Python, son: Máquina de Soporte Vectorial, Naïve Bayes y Naïve Bayes Multinomial.
- Fase de prueba: Los datos de prueba son clasificados de acuerdo al modelo propuesto por el clasificador. La polaridad propuesta a cada opinión de los datos de prueba son comparadas con los datos del gold estándar.
- Evaluación: Para medir los resultados de los experimentos se utiliza como medidas de evaluación *Precisión* y *Recall* para medir los resultados del algoritmo utilizando el clasificador SVM, los resultados del clasificador Naïve Bayes Multinomial, y la medida de *Acurracy* se utiliza para los resultados del algoritmo con el clasificador Naïve Bayes.

Precisión y *Recall* son medidas que se basan en la comparación de un resultado esperado y el resultado efectivo del sistema evaluado [4]. Estas medidas han sido adaptadas para la evaluación de la clasificación en el análisis de sentimientos. La precisión se mide con la Ecuación (1) y el *recall* con la Ecuación (2):

$$Precision = \frac{t_p}{t_p + f_p}, \tag{1}$$

$$Recall = \frac{t_p}{t_p + f_n}, \tag{2}$$

Donde:
t_p: verdaderos positivos,

t_n: verdaderos negativos,
f_p: falsos positivos,
f_n: falsos negativos.

La medida armónica entre el *recall* y la *precisión* es la función F_1 (ver Ecuación 3):

$$F_1 = \frac{2pr}{(p+r)}, \tag{3}$$

Donde:
p: $Precision$,
r: $Recall$.

La medida **Accuracy** (exactitud) se refiere a la evaluación del sesgo de las predicciones, es decir, responde a la pregunta, ¿Cuál es el promedio de las predicciones correctas? La ecuación de la exactitud se muestra en la fórmula (4):

$$Accuracy = \frac{t_p + t_n}{t_p + t_n + f_p + f_n}. \tag{4}$$

A continuación se presentan los resultados obtenidos con el enfoque propuesto.

5. Resultados obtenidos

Esta sección describe los datos que se usaron para probar el enfoque propuesto, también se muestran los resultados obtenidos aplicando lo mencionado anteriormente para cada conjunto de datos.

5.1. Conjunto de datos

Los datos utilizados para esta investigación se tomaron de los conjuntos de datos proporcionados por Semeval 2016 para la solución de la tarea 5, subtarea 2 [9]. Se trabajó con 3 conjuntos diferentes: Conjunto de opiniones para Restaurantes en el idioma español, conjunto de opiniones para Restaurantes en el idioma inglés y conjunto de opiniones para Laptops en el idioma inglés. En la Tabla 1 se muestran ejemplos de opiniones recuperadas de los datos de prueba para cada dominio.

En la Tabla 2 se muestra la cantidad de opiniones en los datos de entrenamiento y en los datos de prueba por cada dominio. En la Tabla 3 se presenta el total de opiniones por tipo de polaridad: positivo (Pos), negativo (Neg), neutral (Neu) y conflicto (Con), en cada dominio de los datos de entrenamiento.

5.2. Resultados experimentales

El objetivo de este artículo es comparar el uso de clasificadores en distintos entornos, Weka y Python, probando los mismos conjuntos de datos en cada uno.

Tabla 1. Ejemplos de opiniones para cada dominio.

Dominio	*Opiniones*
Restaurantes (Español)	La comida estuvo muy sabrosa. Quien sea amante de la carne tiene una carta bastante amplia para elegir., aunque ayer no tenían chuleton. Lo único que nos sorprendió es que nos sirvieran los entrantes y los platos principales a la vez.
Restaurantes (Inglés)	Yum!Serves really good sushi. Not the biggest portions but adequate. Green Tea creme brulee is a must! Don't leave the restaurant without it.
Laptops (Inglés)	Well, my first apple computer and I am impressed. Works well, fast and no reboots. Waiting to install MS Office and see how it goes from there. Have always been a PC guy, but decided to try Apple. Glad I did so far.

Tabla 2. Total de opiniones por conjunto de datos y dominio.

Dominio	*Entrenamiento*	*Prueba*	*Gold*
Restaurantes (Español)	2,121	881	881
Restaurantes (Inglés)	1,435	404	404
Laptops (Inglés)	2,082	545	545

Tabla 3. Conjunto de datos de entrenamiento por polaridad.

Dominio	*Pos*	*Neg*	*Neu*	*Con*	*Total*
Restaurantes (Español)	1,519	443	101	58	2,121
Restaurantes (Inglés)	1,012	327	55	41	1,435
Laptops (Inglés)	1,210	707	123	41	2,081

Para medir los resultados en estos experimentos se utiliza como medidas de evaluación *Precisión* y *Recall* para medir los resultados de SVM y Naïve Bayes Multinomial, y la medida de *Acurracy* usado únicamente para los resultados obtenidos con el clasificador Naïve Bayes.

Resultados en Python. Para el caso del clasificador automático Máquina de Soporte Vectorial (SVM) en Python, el mejor resultado se logra con el conjunto de datos del dominio de Restaurantes en Español obteniendo el 69% de precisión, de la misma manera en el caso del clasificador Naïve Bayes Multinomial obteniendo el 67% de precisión. En el caso de Naïve Bayes se logra el 72% de exactitud.

En las Tablas 4-6 se muestran con detalle los resultados obtenidos en cada clasificador probado y en cada conjunto de datos.

Resultados en Weka. Probando los datos en el entorno de Weka, los resultados obtienen un menor porcentaje de *Acuraccy* en el dominio de Laptops en el idioma inglés. Sin embargo, para el conjunto de Restaurantes en el idioma español se alcanza el 0.69% de precisión con el clasificador Naïve Bayes Multinomial y

Tabla 4. Resultados obtenidos en Python y SVM.

Dominio	*Precisión*	*Recall*	F_1
Restaurantes (Español)	0.69	0.78	0.73
Restaurantes (Inglés)	0.69	0.76	0.72
Laptops (Inglés)	0.66	0.72	0.68

Tabla 5. Resultados obtenidos en Python y Naïve Bayes Multinomial.

Dominio	*Precisión*	*Recall*	F_1
Restaurantes (Español)	0.67	0.73	0.69
Restaurantes (Inglés)	0.64	0.71	0.67
Laptops (Inglés)	0.66	0.71	0.68

Tabla 6. Resultados obtenidos en Python y Naïve Bayes.

Dominio	*Acuraccy*
Restaurantes (Español)	0.72
Restaurantes (Inglés)	0.70
Laptops (Inglés)	0.55

SVM. A continuación se muestran las Tablas 7, 8 y 9 con los resultados obtenidos para cada conjunto de datos.

Tabla 7. Resultados obtenidos en Weka y SVM.

Dominio	*Precisión*	*Recall*	F_1
Restaurantes (Español)	0.69	0.77	0.72
Restaurantes (Inglés)	0.67	0.75	0.70
Laptops (Inglés)	0.60	0.66	0.63

Tabla 8. Resultados obtenidos en Weka y Naïve Bayes Multinomial.

Dominio	*Precision*	*Recall*	F_1
Restaurantes (Español)	0.69	0.77	0.73
Restaurantes (Inglés)	0.70	0.76	0.73
Laptops (Inglés)	0.67	0.66	0.66

Tabla 9. Resultados obtenidos en Weka y Naïve Bayes.

Dominio	*Acuraccy*
Restaurantes (Español)	0.71
Restaurantes (Inglés)	0.59
Laptops (Inglés)	0.40

6. Comparación de resultados entre Python y Weka

Teniendo los resultados con cada entorno y clasificador, con base en los resultados de la medida armonica F_1 y *accuracy* se observa que SVM y Naïve Bayes tienen un mejor comportamiento en Python, y con Naïve Bayes Multinomial el algoritmo se comporta mejor en Weka (ver Tabla 10). Consideramos que se obtiene un mejor comportamiento de aprendizaje automático utilizando Python, puesto que la extracción de características se realizan automáticamente usando las herramientas para el análisis de datos y procesamiento del lenguaje natural *scikit-learn* y *NLTK*.

Tabla 10. Comparación entre Python y Weka.

Dominio	SVM		NB Multinomial		Naïve Bayes	
	Weka	Python	Weka	Python	Weka	Python
	F_1	F_1	F_1	F_1	*Accuracy*	*Accuracy*
Restaurantes (Español)	0.72	**0.73**	**0.73**	0.69	0.71	**0.72**
Restaurantes (Inglés)	0.70	**0.72**	**0.73**	0.67	0.59	**0.70**
Laptops (Inglés)	0.63	**0.68**	0.66	**0.68**	0.40	**0.55**

En las Figuras 1, 2 y 3 se muestran las gráficas con los resultados de los experimentos entre ambos entornos y tipo de clasificador.

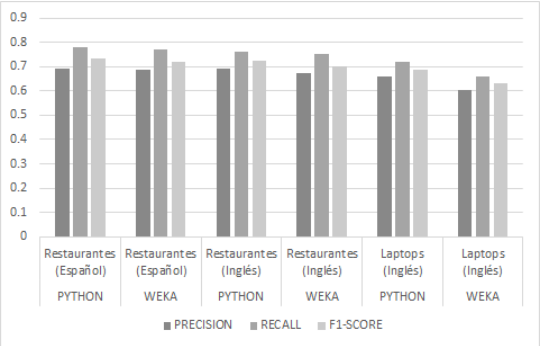

Fig. 1. Resultados con Máquina de Soporte Vectorial.

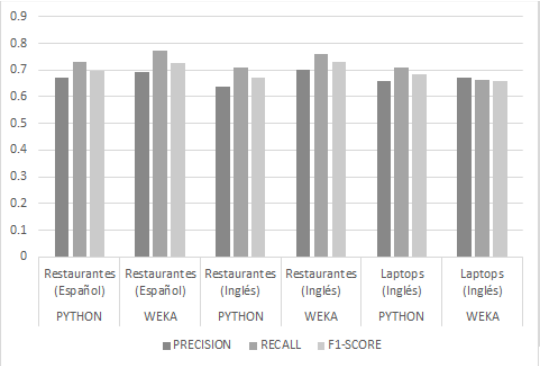

Fig. 2. Resultados con Naïve Bayes Multinomial.

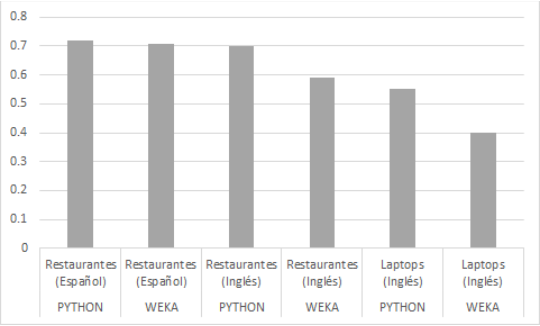

Fig. 3. Resultados con Naïve Bayes.

7. Conclusiones

En esta investigación se presentan los resultados obtenidos en el análsis de sentimientos utilizando tres clasificadores de aprendizaje supervisado: Máquina de Soporte Vectorial, Naïve Bayes y Naïve Bayes Multinomial. Estos algoritmos son utilizados para clasificar opiniones de los dominios de Restaurantes y Laptops, por cada opinión se detecta una de las cuatro posibles polaridades: positivo, negativo, neutral y conflicto. También se utiliza Python con *nltk* y *scikit learn*, y Weka. Con base en los resultados obtenidos se muestra que los mejores resultados son con los clasificadores diseñados en Python. Logrando un 69% de precisión para el dominio de Restaurantes en Español con el clasificador SVM en Python, el 70% de precisión en el dominio de Restaurantes en Inglés con el clasificador Naïve Bayes Mutinomial en Weka y el 67% de precisión para el dominio de

Laptops en Weka con Naïve Bayes Multinomial. En el caso de la exactitud en los tres dominios son mejores los resultados en Python. Consideramos que se tiene este comportamiento debido a que la extracción de características, en el caso de Python, son con las herramientas interconstruidas en el lenguaje. Así mismo se concluye que con las pruebas realizadas en este trabajo de investigación el mejor clasificador es SVM con Python, Naïve Bayes Multimonial con Weka y Naïve Bayes con Python. Como trabajos a futuro consideramos el uso de otros tipos de características así como el empleo de herramientas de medición de polaridad de las palabras como SentiWordNet.

Agradecimientos. Esta investigación es parcialmente apoyada por el proyecto PRODEP-SEP ID 00570 (EXB-792) DSA/103.5/15/10854, por el proyecto ID 00570 VIEP-BUAP. Apoyado por el Fondo Sectorial de Investigación para la Educación, proyecto Conacyt 257357.

Referencias

1. Bermejo, P., Gámez, J.A., Puerta, J.M.: Improving the performance of naive bayes multinomial in e-mail foldering by introducing distribution-based balance of datasets. Expert Systems with Applications 38(3), 2072 – 2080 (2011), http://www.sciencedirect.com/science/article/pii/S0957417410007748
2. Betancourt, G.A.: Las máquinas de soporte vectorial (svms). Scientia Et Technica 27 XI, 67–72 (2005)
3. Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., VanderPlas, J., Joly, A., Holt, B., Varoquaux, G.: API design for machine learning software: experiences from the scikit-learn project. In: ECML PKDD Workshop: Languages for Data Mining and Machine Learning. pp. 108–122 (2013)
4. Euzenat, J.: Semantic precision and recall for ontology alignment evaluation. In: IJCAI. pp. 348–353 (2007)
5. Gamallo, P., Garcia, M.: Citius: A naive-bayes strategy for sentiment analysis on english tweets. In: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). pp. 171–175. Association for Computational Linguistics and Dublin City University, Dublin, Ireland (August 2014)
6. Hridoy, S.A.A., Ekram, M.T., Islam, M.S., Ahmed, F., Rahman, R.M.: Localized twitter opinion mining using sentiment analysis. Decision Analytics 2(1), 1 (2015)
7. Mulay, S.A., Joshi, S.J., Shaha, M.R., Vibhute, H.V., Panaskar, M.P.: Sentiment analysis and opinion mining with social networking for predicting box office collection of movie. International Journal of Emerging Research in Management & Technology 5(1), 74–79 (2016)
8. Narayanan, V., Arora, I., Bhatia, A.: Fast and accurate sentiment classification using an enhanced naive bayes model. Lecture Notes in Computer Science 8206, 194–201 (2013)
9. Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S., AL-Smadi, M., Al-Ayyoub, M., Zhao, Y., Qin, B., De Clercq, O., Hoste, V., Apidianaki, M., Tannier, X., Loukachevitch, N., Kotelnikov, E., Bel, N., Jiménez-Zafra, S.M., Eryiğit, G.: Semeval-2016 task 5: Aspect based sentiment analysis. In: Proceedings of the 10th International Workshop on Semantic Evaluation

(SemEval-2016). pp. 19–30. Association for Computational Linguistics, San Diego, California (June 2016), http://www.aclweb.org/anthology/S16-1002
10. Poria, S., Cambria, E., Howard, N., Huang, G.B., Hussain, A.: Fusing audio, visual and textual clues for sentiment analysis from multimodal content. Neurocomputing 174, 50–59 (2016)
11. Rennie, J.D., Shih, L., Teevan, J., Karger, D.R., et al.: Tackling the poor assumptions of naive bayes text classifiers. In: Proceedings of the Twentieth International Conference on Machine Learning. vol. 3, pp. 616–623. Washington DC) (2003)
12. Rish, I.: An empirical study of the naive bayes classifier. In: IJCAI 2001 workshop on empirical methods in artificial intelligence. vol. 3, pp. 41–46. IBM New York (2001)
13. Sangeetha Suresh Harikantra, R.F.: Opinion mining on twitter data. International Journal of Innovative Research in Science, Engineering and Technology 5(9), 205–209 (2016)
14. Yadav, V.: thecerealkiller at semeval-2016 task 4: Deep learning based system for classifying sentiment of tweets on two point scale. In: Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016). pp. 100–102. Association for Computational Linguistics, San Diego, California (June 2016)

ISSN 1870-4069

Exploración sobre el máximo desempeño en la selección no supervisada de términos para agrupamiento de textos

Héctor Jiménez-Salazar

Universidad Autónoma Metropolitana, Departamento de Tecnologías de la Información, Unidad Cuajimalpa, México

hgimenezs@gmail.com

Resumen. El agrupamiento de textos es uno reto importante por la diversidad de aplicaciones que se derivan de la solución de dicha tarea. Un elemento indispensable en el agrupamiento es la selección de términos para representar lo mejor posible los textos. Aunque hay muchos métodos orientados a extraer términos de documentos para llevar a cabo categorización de textos, son pocos los que enfrentan la tarea de agrupamiento por la dificultad que se presenta al no contar con la clase de cada uno de los documentos. En este trabajo se propone un nuevo método que extrae los términos para representar los textos y, al ser agrupados, se obtiene el desempeño máximo en una cantidad notable de casos. Las pruebas se llevaron a cabo con un conjunto de varias decenas de colecciones de textos cortos (tuits), lo cual permite observar el comportamiento del método. El planteamiento que subyace al método está basado en el ascenso máximo de la similitud de los documentos y en las propiedades de unificación y diversificación de los términos expuestas por G. Zipf.

Palabras clave: Selección no supervisada de términos, agrupamiento de textos, tuits.

Exploration on the Maximum Performance of Unsupervised Term Selection for Text Clustering

Abstract. Text clustering is a major challenge for a diversity of applications derived from the solution of this task. An indispensable element in the text clustering is the selection of terms in order to get the best representation of texts. Although there are many methods designed to extract terms from documents to carry out categorization of texts, there are few methods that face the task of clustering, due to the difficulty presented by not having the class of each document. In this paper a new method is proposed that extracts the terms to represent the texts and, being grouped, it obtains the maximum performance in a high number of cases. Tests were conducted with a set of several tens of collections composed by short texts (tweets), which allows to know the behavior

of the method. The approach underlying of the method is based on maximum rise of the similarity of documents and properties of unification and diversification of the terms given by G. Zipf.

Keywords. Unsupervised term selection, text clustering, tweets.

1. Introducción

El agrupamiento de textos es uno reto importante no solo por la diversidad de aplicaciones que se derivan de la solución de dicha tarea, como la segmentación de textos [6], la inducción de sentidos [1], y la visualización [7], entre otras, sino además por la alta demanda que hay en la actualidad debido a los volúmenes crecientes de texto en línea que requieren sistematización para el aprovechamiento de su contenido [4].

Dada una colección de textos, el problema de agrupamiento establece como meta reunir en grupos a los textos que satisfagan mayor similitud entre los del mismo grupo y menor similitud entre textos de grupos diferentes. Para resolver este problema se representan los textos mediante algún criterio (bolsa de palabras, vectorial, distribucional, etc.) y se aplica un método de agrupamiento (K-means, k-NN, etc.). La evaluación de la efectividad del agrupamiento puede hacerse con una colección de textos clasificados manualmente o *gold standard*, de tal manera que los grupos obtenidos puedan compararse con el *gold standard*.

Uno de los factores que influye de manera crucial en el resultado del agrupamiento es la representación de los textos; por ejemplo, es conocido que si se incluyen todos los términos de los textos de la colección, muchos de ellos resultarán "ruidosos": sesgarán la construcción de los grupos, incluyendo o excluyendo textos en forma incorrecta. Así, se concibe representar los textos utilizando los términos que logran mejor desempeño en la tarea de agrupamiento. Éste es un problema de optimización combinatoria que por su alta dimensionalidad (tamaño del vocabulario) se ha enfrentado con un enfoque intuitivo a través de dos pasos: definir un criterio de importancia de los términos y, con otro criterio, elegir una parte de los términos que mejor representación hacen de los textos a agrupar.

Por tanto, debe adoptarse un criterio que asigne la importancia a cada término, usando un valor numérico y, finalmente, tomar una parte de los términos más importantes. Los métodos de selección de términos, entonces, deberán proporcionar un ordenamiento de los términos según su importancia (*ranking*) y un criterio de selección, es decir, la cantidad de términos que se tomarán de la lista ordenada para representar solamente con ellos todos los textos de la colección.

Es importante señalar que la tarea de agrupamiento textual difiere de la tarea de categorización textual. En el primer caso las aplicaciones de los métodos se limitan a un conjunto de textos sin ninguna información sobre la clase de ellos y justamente se trata de definir la clase de cada uno de los textos, mientras que en el segundo caso se cuenta con textos y la clase a la cual pertenecen, a partir de lo cual se espera determinar a cual de las clases definidas pertenece uno o varios nuevos textos.

Al resolver el problema de agrupamiento, no se tiene ninguna información previa. Puede ser, incluso, que tengamos una colección de textos de la cual tenemos clasificada una parte y se desee agrupar el resto, sin considerar las clases conocidas; esto es, no se podrá tomar en cuenta la importancia de los términos observada en la parte clasificada de los textos.

Por ejemplo, en la figura 1 aparecen dos líneas, éstas representan la efectividad del agrupamiento de dos colecciónes del mismo dominio (autos). Cada curva se define por puntos que consideran, en el eje horizontal, un múltiplo del 5% del total de términos y, en el vertical, la efectividad del agrupamiento. Lo que se observa es que si tomamos como referencia el porcentaje para el cual se obtuvo máximo desempeño en la colección 13 para usarlo en la selección de términos de la colección 9, tendremos un gran fracaso. En conclusión, en el agrupamiento no basta que un método que determina la importancia de los términos sea superior a otros, además deberá contarse con una forma de determinar cuántos términos elegir.

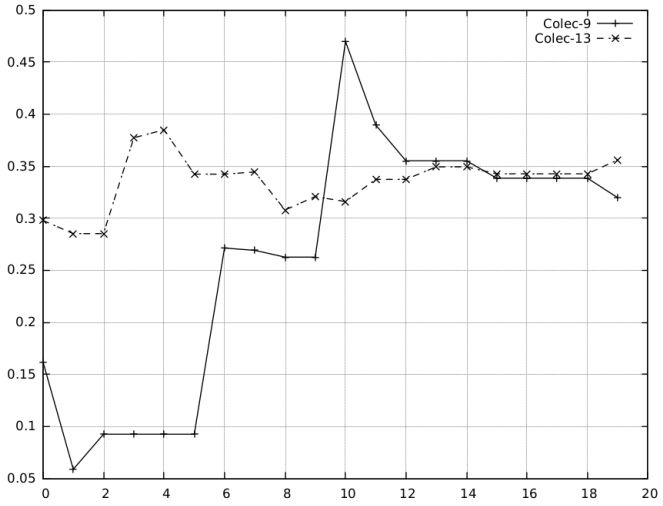

Fig. 1. Desempeño de un método de selección en dos colecciones.

Este trabajo expone algunos experimentos realizados sobre agrupamiento de textos en un conjunto de colecciones con el fin de analizar las regularidades de un método de selección de términos y otro de pesado de términos (que permite hacer el *ranking*). Solamente se usa un método de agrupamiento: *K-star* (una variante de *K-Nearest Neighbor*) [10], el cual determina en forma heurística el número de grupos. El presente trabajo aporta resultados orientados hacia el conocimiento sobre la representación de textos para mejorar la tarea de agrupamiento a través de:

1. Un método para pesado de términos.

2. Un criterio para seleccionar los términos pesados.

lo anterior sin considerar ninguna información previa: umbrales de selección para otras colecciones, información sobre indicadores en una parte clasificada de la colección, etc.

En este trabajo se utilizó una parte de la colección de RepLab-2013, competencia internacional sobre monitoreo de reputación (organizaciones, personalidades, etc.) y los detalles sobre ella se exponen en la sección 2. Si bien se propone y explora un método de selección, se utilizan como contraste otros métodos de pesado de términos, todo esto se describe en la sección 3. La sección 4, por su parte, contiene el procedimiento seguido para realizar el experimento y los resultados obtenidos. Las conclusiones de este trabajo aparecen en la última sección.

2. Descripción de las colecciones de prueba

En los experimentos se utilizó el conjunto de datos de la competencia RepLab 2013 [2]. Para las tareas previstas en RepLab-2013 este conjunto de datos fue anotado manualmente con (a) pertenencia o no a la entidad, (b) tema, (c) polaridad, y (d) grado de prioridad. De estas anotaciones solamente interesa el tema, y se usa con el fin de medir el desempeño de los métodos propuestos.

Este conjunto de datos está compuesto por cuatro dominios: autos, bancos, universidades, y música. En cada uno de los dominios hay entidades, y por cada entidad varios temas. Finalmente cada tema contiene tuits. En total son 61 colecciones de texto. Además, en todas las colecciones aparecen tuits que no tienen ninguna relación con el dominio o tema.

Como se sabe, cada tuit contiene información que puede ser de importancia para tareas diversas, como el lenguaje usado, *hashtag*, etc. En nuestros experimentos, de esta información se toma en cuenta solamente el lenguaje para dividir en dos las colecciones: inglés y español.

Con la anterior estructura se tienen dos conjuntos de datos, de entrenamiento y prueba. Nuestros experimentos trabajan únicamente con la colección de entrenamiento en español. Los tuits fueron preprocesados eliminando las palabras cerradas, asimismo, los enlaces dentro de los tuits y nombres de usuario fueron removidos, pero la conversación derivada de cada tuit se conservó como parte del tuit inicial.

Después de preprocesar las colecciones se obtuvo el material al cual se aplicaron los métodos que se presentan en la sección 3.. La tabla 1 muestra los siguientes valores: en la primera columna, el número total de colecciones; en la segunda, el número promedio de clases por colección; en la tercera, el número promedio de textos por colección; y en la última, el número promedio de palabras que ocurren por colección.

Tabla 1. Composición de las colecciones de prueba utilizadas.

Colecciones	Prom.clases	Prom.textos	Prom.palabras
54	11.20	117.27	1288.07

3. Métodos de selección utilizados

En esta sección se presenta la base sobre la que se apoya el método de ponderación de términos propuesto. Asimismo, se describen los métodos de pesado de términos que fueron analizados. El método que a continuación se presenta fue utilizado en la competencia RepLab13 [2,9] (UAMCLyR-7); aunque solamente en el presente trabajo aparece la descripción detallada del método y nuevos experimentos que muestran su grado de efectividad.

3.1. Método propuesto

El método surge de la relación que guarda el promedio de la similitud entre todas las parejas de documentos de una colección con la entropía [3]. Esta propiedad puede enunciarse como: a mayor similitud menor entropía; es decir, los términos que componen los documentos son más informativos, cuando la entropía es menor. Relacionado con lo anterior se consideran dos conceptos de G. Zipf, *diversificación* y *unificación* [13].

Para trabajar con estas ideas considérese una colección de documentos, $\mathcal{C} = \{d_1, \ldots, d_n\}$. Los términos de $\mathcal{C}$ manifiestan su propiedad unificadora a través de una alta similitud entre los documentos, y su propiedad diversificadora mediante baja similitud entre los elementos de $\mathcal{C}$. Un ejemplo de términos diversificadores son los *hapax*, y términos muy frecuentes entre documentos serán unificadores.

Se representa cada término t de $\mathcal{C}$ por el conjunto de clases que contienen a t, $\bar{t}$. Un término es diversificador si $\#\bar{t}$ es bajo, y es unificador si $\#\bar{t}$ es alto. Ciertamente ambos tipos de términos son necesario en la tarea de agrupamiento.

El enfoque seguido está basado en la cuantificación de la propiedad unificadora de los términos, a través de la fórmula:

$$U(t_i) = \frac{1}{r} \sum_{j \neq i} sim(\bar{t_i}, \bar{t_j}),$$

donde $r = \#\{t_j | sim(\bar{t_i}, \bar{t_j}) \neq 0)\}$, y sim es una medida de similitud. También es útil considerar:

$$S(\mathcal{C}) = \frac{2}{n(n-1)} \sum_{i \neq j} sim(d_i, d_j).$$

$S(\mathcal{C})$ da un valor global sobre la unificación a partir de los términos que componen los documentos de $\mathcal{C}$. Faltaría explicar cómo elegir un conjunto de términos que tenga una proporción adecuada de ambos términos, unificadores y diversificadores.

Como un primer paso de la representación de documentos podemos incluir la mayor parte de los términos diversificadores, lo cual se consigue eligiendo términos con los valores más bajos de U. Llamemos V a este conjunto, y el $p\%$ de los términos con los valores más bajos de U será denotado por V_p. Dado p, se calcula $S(\mathcal{C}_p)$, donde $\mathcal{C}_p$ tiene los mismos documentos que $\mathcal{C}$ pero representados unicamente por sus términos que pertenecen a V_p. Cuando p crece, los valores de U también, pero para cierto porcentaje q, $\mathcal{C}_q$ se satura con términos unificadores provocando un gran descenso en el valor $S(\mathcal{C}_q)$. Este descenso se interpreta como un indicador de una selección balanceada con ambos tipos de términos.

En las colecciones de prueba se observó correlación entre el valor máximo de F (usando precisión y exhaustividad) y el descenso abrupto de $S(\mathcal{C})$. Este descenso está asociado con la similitud máxima de los documentos. En suma, el método sigue dos pasos:

1. Determinar un conjunto balanceado de términos usando U y S:
   (a) Calcular $U(t)$ para todos los términos de $\mathcal{C}$ y ordenar en forma creciente: $T_U = [U(t_1), \ldots, U(t_n)]$.
   (b) Dividir T_U en m partes, para proveer m conjuntos de términos: $V_i, (1 \leq i \leq m)$ representa las primeras i partes de términos (en el experimento se usó $m = 10$).
   (c) Calcular $S(\mathcal{C}_i)$, correspondiente a cada conjunto de selección V_i y determinar el punto de máximo descenso; j.
2. Aplicar el algortimo de agrupamiento a $\mathcal{C}_j$.

Notemos que en el anterior procedimiento puede usarse cualquier otro método de pesado. El que se ha presentado, definido por U, será llamado DU en lo sucesivo.

Por otro lado, las clases usadas para representar cada término ($\bar{t}$), fueron determinadas usando el resultado de la aplicación del algoritmo *K-Star* sin seleccionar términos de los documentos del mismo corpus de trabajo.

3.2. Otros métodos de selección

Adicionalmente al método DU, se utilizaron tres métodos de selección de términos: TA (*term average*), DF (*document frequency*), y TC (*term contribution*).

En primer lugar se define el peso *frecuencia en documentos*, DF, como $DF(t_i) = |\{d_j | t_i \text{ ocurre en } d_j\}|$ [12]. DF se considera en el el cálculo de un peso muy utilizado en recuperación de información [8] y su normalización:

$$tfidf_{ij} = f_{ij} \cdot \log\left(\frac{2 * n}{DF(t_i)}\right), \qquad tfin_{ij} = \frac{tfidf_{ij}}{\sqrt{\sum_{k=1}^{n} tfidf_{ik}^2}},$$

donde f_{ij} es la frecuencia del término t_i en el documento d_j. Con lo anterior se define el peso *contribución del término* t_i [5]:

$$TC(t_i) = \sum_{i=1}^{n} \sum_{j=1, j\neq i}^{n} tfin_{ij} \cdot tfin_{ik}.$$

Y el peso *promedio del término* basado en $tfidf_{ij}$, TA [11]:

$$TA(t_i) = \frac{1}{n} \sum_{j=1}^{n} tfidf_{ij}.$$

A continuación se describe la forma en que se midió el desempeño de los métodos de selección utilizados en el agrupamiento. Dada una colección $\mathcal{C}$ con m clases $C_1, \ldots, C_m$ y un agrupamiento $\mathcal{G} = \{G_1, \ldots, G_s\}$, con base en las siguientes medidas $P_{ij} = \frac{|Ci \cap Gj|}{|Ci|}$, $R_{ij} = \frac{|Ci \cap Gj|}{|Gi|}$, $1 \leq i \leq m$ y $1 \leq j \leq s$, se define la medida $F_{ij} = \frac{2 \cdot Pij \cdot Rij}{Pij + Rij}$ de la clase C_i con respecto al grupo G_j. Con ello se calcula una medida global de todo el agrupamiento $\mathcal{G}$ como:

$$F = \sum_{i=1}^{m} \frac{|C_i|}{|\mathcal{C}|} F_i,$$

donde $F_i = \max\{F_{ij}\}_{j=1}^{m}$. Con esta medida se realizó la evaluación de los agrupamientos realizados.

4. Experimentos y resultados

De acuerdo con lo expuesto en la sec. 3., el primer paso fue calcular los pesos por cada uno de los métodos utilizados para ordenar términos por su importancia; TA, TC, DF y DU. Las listas de términos, ordenados de mayor a menor según su peso (excepto DU, que es de menor a mayor), se utilizan para tomar porcentajes crecientes, desde 5%, e incrementando 5% hasta tener la totalidad de términos. Representando cada documento con únicamente los términos seleccionados se llevó a cabo el agrupamiento con *K-star*. El método de agrupamiento calcula la similitud promedio entre todos los textos y utiliza este valor como criterio para decidir si dos textos pertenecen a la misma clase: se toma como la similitud mínima que deben satisfacer textos de un mismo grupo. Como se verá, el ascenso de la similitud promedio entre los documentos es un indicador de una buena selección en ciertas condiciones.

Ya que se cuenta con 54 colecciones de textos, un primer experimento fue conocer el comportamiento de los métodos de pesado de términos. Como ejemplo en la figura 2 se muestran los resultados de la selección obtenida por el criterio de similitud en una muestra de seis colecciones. Cada punto del eje horizontal corresponde a una colección, y la altura al desempeño obtenido por uno de los métodos.

En esta gráfica se observa la variabilidad de los valores de F para varias colecciones y no parece haber regularidad de un método a través de las colecciones. Por ejemplo, para las primeras dos colecciones el método TC resulta mejor, no así en las demás colecciones. Algo semejante sucede si exploramos la totalidad de las colecciones.

Por otro lado, en la figura 3 se observa el máximo desempeño que obtuvo cada método en las seis colecciones. En este caso sí se tiene que el método TA

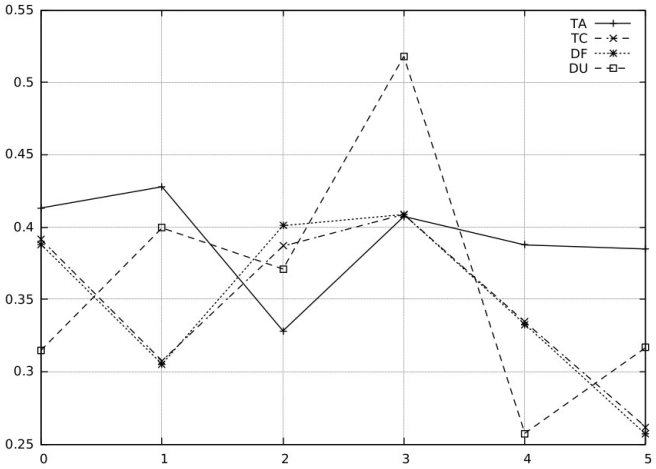

Fig. 2. Valor de F obtenido con el criterio de similitud máxima sobre una muestra de seis colecciones.

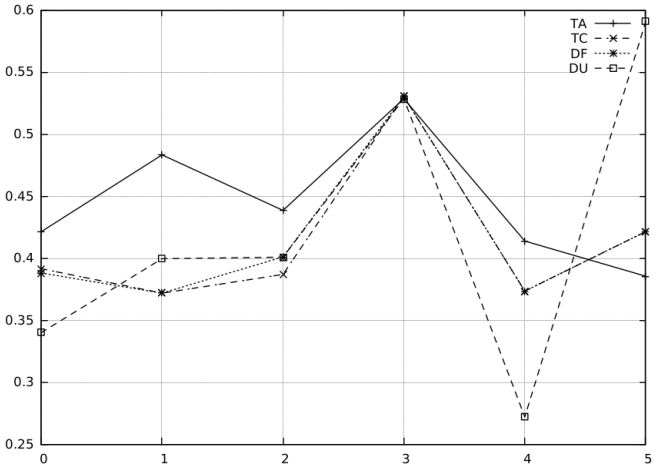

Fig. 3. Máximo valor de F que puede obtenerse con cada método de selección sobre una muestra de seis coleciones.

supera en la mayoría de casos. Sin embargo, no tenemos acceso a esta información pues requiere del conocimiento de la clase de cada texto. Se ha comentado que el porcentaje de términos de un ordenamiento, dado por un método de ponderación, que obtuvo buen desempeño en una colección no es útil en otras, aún cuando se observe un mismo porcentaje conveniente para varias colecciones, no puede este hecho asegurar que funcionará tal porcentaje en una nueva colección.

Para discernir sobre el mejor desempeño de los métodos de selección se

diseño un experimento para conocer la relación entre el desempeño obtenido y el máximo posible que podía alcanzar cada método: qué tan cerca están estos dos valores.

El experimento consistió en efectuar el agrupamiento de cada una de las colecciones utilizado los métodos de selección TA, TC, DF y DU. Además se aplicó el criterio de similitud máxima para determinar la selección que presumiblemente obtendría el mejor desempeño.

Enseguida se exponen los pasos realizados para encontrar los valores que permiten comparar el desempeño de los métodos utilizando el criterio de similitud máxima. Denotaremos un método de selección por M.

1. Para cada una de las colecciones, $\mathcal{C}_i$ $(1 \leq i \leq 54)$:
   (a) Usando cada selección de términos S_{Mij} $(1 \leq j \leq 20)$, proporcionada por el método M, se representaron los textos de $\mathcal{C}_i$ y se realizó el agrupamiento de $\mathcal{C}_{ij}$.
   (b) En cada agrupamiento se obtuvo el desempeño: F_{Mij}.
   (c) De los valores de desempeño se obtuvo el valor promedio y máximo: $\bar{F}_{Mi}$, $F_{Mi,max}$.

En totalidad se tienen 54 parejas $\bar{F}_{Mi}$, $F_{Mi,max}$ para cada método. Con ellas se llevó a cabo una prueba de hipótesis para conocer en qué medida el método M obtenía un valor cercano al máximo. La prueba entonces utliza una muestra $(n = 54)$ en donde se supone que los valores de $\bar{F}_{Mi}$ y $F_{Mi,max}$ se distribuyen normalmente. Se llevó a cabo la prueba de diferencia de medias:

$$H_0 : \bar{F}_M = \bar{F}_{M,max}$$
$$H_1 : \bar{F}_M \neq \bar{F}_{M,max}$$

donde $\bar{F}_M$ es la media de los valores de F obtenidos con el criterio de similitud máxima para el método M, y $\bar{F}_{M,max}$ es el promedio de los valores máximos de F obtenidos para el método M. La tabla 2 muestra los resultados de las pruebas realizadas con un nivel de significancia del 95%. Las columnas, de izquierda a derecha, corresponden a: el método (M), el promedio de los valores F para M, el promedio de los máximos valores de F que obtuvo M, la diferencia de los anteriores valores, el valor crítico de la normalización de la distribución de la diferencia de valores medios, y la conclusión de la prueba de hipótesis, respectivamente.

Como se aprecia en la tabla 2, el método DU es el que mejor se aproxima al máximo que se puede obtener utlizando el criterio de similitud máxima. Notamos también que los otros métodos, aunque no siguen el criterio de máxima similitud, pueden obtener desempeños altos, lo cual se aprecia en la columna $\bar{F}_{M,max}$. Sin embargo, se ha dicho que no se tiene un acceso seguro a dichos valores en forma no supervisada.

5. Conclusiones

Se ha presentado un nuevo método para pesado de términos, DU, que funciona en combinación con el criterio de ascenso máximo de similitud (la similitud

Tabla 2. Resultados de las pruebas de diferencia entre $\bar{F}_M$ y $\bar{F}_{M,max}$.

M	$\bar{F}_M$	$\bar{F}_{M,max}$	DIFER	V.crítico	Acepta
TA	0.460015	0.556161	0.0961463	0.0724493	H_1
TC	0.427633	0.508341	0.0807074	0.0689402	H_1
DF	0.448352	0.533959	0.0856074	0.0756838	H_1
DU	0.473756	0.498837	0.0250815	0.0846659	H_0

promedio entre documentos que usan una selección de términos). La aplicación del método de selección (pesos y criterio de selección) a 54 colecciones de tuits obtuvo una efectividad muy cercana al máximo posible obtenido por este método. Al utilizar otros métodos de pesado de términos con el mismo criterio de selección se obtuvo una diferencia significativa con respecto a la máxima efectividad posible. Es importante destacar que los otros métodos obtienen un $\bar{F}_{max}$ mayor que el de DU. Esto sugiere, por ejemplo, que si pudiera adaptarse el criterio a estos métodos, su efectividad crecería.

En suma, los métodos TA, TC, y DF son menos compatibles con el criterio de ascenso máximo de similitud que el método DU. Aún cuando la afirmación anterior es estadísticamente válida para el conjunto de colecciones empleada, quizá ello no pueda generalizarse para colecciones semejantes. Ciertamente, habría un sustento para usar el método propuesto en agrupamiento de tuits, pero también es necesario conocer el alcance del método, a través de la realización de más pruebas tanto para textos que no sean tuits, como para textos de mayor tamaño.

Agradecimientos. El autor desea expresar su agradecimiento al Departamento de Tecnologías de la Información de la UAM Cuajimalpa por el apoyo parcial recibido al presente trabajo.

Referencias

1. Agirre, E., Soroa A.: Semeval-2007 task 02: evaluating word sense induction and discrimination systems. ACL (2007)
2. Amigó, E.; Carrillo de Albornoz, J.; Chugur, I.; Corujo, A.; Gonzalo, J.; Martín, T.; Meij, E.; de Rijke, M., Spina, D.: Overview of RepLab 2013: Evaluating Online Reputation Monitoring Systems. CLEF 2013, LNCS 8138, pp. 333–352 (2013)
3. Dash, M., Liu, H.: Feature Selection for Clustering. PAKDD 2000 (2000)
4. Jain, A.; Murty, M., Flynn, P.: Data Clustering: A Review. ACM Computing Surveys 31 (3) (1999)
5. Liu, T.; Lui, S.; Chen, Z., Ma, W.: An Evaluation of Feature Selection for Text Clustering. Proc. of the 20th Int. Conf. on Machine Learning (2003)
6. Lu, Q.; Conrad, J.; Al-Kofahi, K., Keenan, W.: Legal document clustering with built-in topic segmentation. Proc. of the 20th ACM Int. Conf. on Information and Knowledge Management, ACM (2011)

7. Metsalu, T., Vilo, J.: ClustVis: a web tool for visualizing clustering of multivariate data using Principal Component Analysis and heatmap. Nucleic Acids Research (2015)
8. Salton, G., Buckley, C.: Term weighting approaches in automatic text retrieval. Information Processing and Management, 24(5):513–523, issue 5 (1988)
9. Sánchez-Sánchez, C.; Jiménez-Salazar, H., Luna-Ramírez, W.: UAMCLyR at Replab2013: Monitoring Task. Notebook for RepLab at CLEF (2013)
10. Shin, K., Han, S.: Fast clustering algorithm for information organization. Proc. of the CICLing 2003 Conf. Volume 2588 of LNCS Springer-Verlag (2003)
11. Tang, B.; Shepherd, M.; Milos, E., Heywood, M.: Comparing and combining dimension reduction techniques for efficient text clustering. Int. Workshop on Feature Selection for Data Mining (2005)
12. Yang, Y., Pedersen, J. A.: Comparative Study of Feature Selection in Text Categorization. Proc. of SIGIR (1995)
13. Zipf, G.: Human behavior and the principle of least effort. Cambridge, (Mass.), Addison-Wesley (1949)

ISSN 1870-4069

Método de corrección ortográfica basado en la frecuencia de las letras

Edgar Moyotl-Hernández

Benemérita Universidad Autónoma de Puebla
Facultad de Ciencias Físico Matemáticas, Puebla, México

emoyotl@fcfm.buap.mx

Resumen. En este trabajo se presentan los primeros resultados de un método de corrección ortográfica para la variante del español mexicano, basado en la distancia de edición propuesta por Levenshtein. Para mejorar el funcionamiento de este algoritmo, se propone asignar costos diferentes a las operaciones de edición tomando en cuenta la frecuencia de las letras. Los resultados obtenidos en la evaluación son satisfactorios, especialmente si se considera que se trata de un corrector ortográfico de propósito general y que las palabras se analizan sin ningún tipo de información contextual. Además, este enfoque es capaz de detectar errores que otros correctores no identifican.

Palabras clave: Corrector ortográfico, errores ortográficos, distancia de edición, algoritmo de Damerau-Levenshtein.

Spell Checking Method Based on the Frequency of Letters

Abstract. This paper presents the first results of a spell checking method for variant of Mexican Spanish based on the Levenshtein edit distance. To improve the performance of this algorithm, we propose to assign different costs to editing operations taking into account the frequency of the letters. The results of the evaluation are good, especially if you consider that this algorithm is a spell checker general purpose and that the words are analyzed without any contextual information. In addition, this approach detects errors that other spelling checkers are not able to identify.

Keywords. Spell checker, orthographic errors, edit distance, Damerau-Levenshtein algorithm.

1. Introducción

Con el aumento en la cantidad de información textual generada por las personas y disponible en formato digital, se requieren herramientas que procesen

la mayor cantidad de información posible. Sin embargo, las personas en su redacción producen un sin número de errores y la presencia de estos errores en los textos reduce las posibilidades de éxito a las aplicaciones estándar de *Procesamiento de Lenguaje Natural (PLN)*, encargadas del análisis y procesamiento de documentos. Por ejemplo, un buscador de textos no podrá recuperar documentos para una consulta realizada por un usuario si la misma se escribió mal o los documentos contienen errores en las palabras con las que se está consultando.

Los algoritmos de corrección tienen como objetivos: detectar y corregir, ya sea de forma automática o interactiva, los errores de redacción generados por los humanos; esto con el fin de aumentar la calidad de los textos. Estos algoritmos son utilizados ampliamente por los procesadores de texto, y también se aplican en tareas como la normalización de textos, detección de plagios, reconocimiento óptico de caracteres, recuperación de información entre otras. Por ejemplo, algunos sistemas buscadores como Google[1] son capaces de detectar errores en las consultas mal escritas y brindar sugerencias de palabras correctas; si se teclea *"corector"* en la barra de búsqueda y se da la orden de buscar, Google instantáneamente mostrará *"Se muestran resultados de* ***corrector****"*.

Justamente, la mayoría de los correctores existentes se enfocan en corregir errores ortográficos; su función es identificar palabras que posiblemente estén mal escritas y presentar al usuario una serie de alternativas de corrección. Para saber si una palabra está escrita correctamente o no, el enfoque más sencillo consiste en utilizar como referencia un diccionario con la lista de (casi) todas las palabras válidas de la lengua tratada. Una palabra está escrita correctamente si se encuentra en el diccionario, en caso contrario es considerada un error. Para este trabajo la composición del diccionario se hizo con el *Corpus de Referencia del Español Actual (CREA)*[2] compuesto de una amplia variedad de textos, producidos en todos los países de habla hispana desde 1975 hasta 2004 y cubre el español mexicano.

Para obtener la lista de alternativas en el caso de que la palabra sea errónea, el corrector buscará en el diccionario las palabras que se obtengan a partir de la palabra incorrecta mediante las operaciones elementales de caracteres: inserción, borrado, sustitución o transposición, las operaciones permitidas en la distancia Damerau-Levenshtein [2], [5]. A diferencia del modelo original, el método propuesto asigna diferentes pesos a las operaciones de acuerdo con la frecuencia de las letras en el español de México[3]. Así mismo, el orden de las alternativas para elegir la mejor candidata se apoya en la probabilidad de ocurrencia en el corpus.

Este trabajo está organizado de la siguiente forma. En la sección 2., se revisan las características de la corrección ortográfica. Luego, en la sección 3. se describe el concepto de distancia de edición y se muestra su uso en los correctores ortográficos. Posteriormente, en la sección 4. se presentan las ideas utilizadas en el método propuesto para modificar el costo de las operaciones de edición.

[1] https://www.google.com
[2] http://www.rae.es/recursos/banco-de-datos/crea
[3] http://dem.colmex.mx

Después en la sección 5., se describen los datos, los experimentos y los resultados de esta propuesta para corregir textos. Finalmente, se presentan las conclusiones y el trabajo futuro que se espera desarrollar.

2. Corrección ortográfica

Los errores de redacción se pueden clasificar de diferentes maneras. En [6] se propone la siguiente clasificación:

- Errores ortográficos (palabras no presentes en el idioma).
- Errores gramaticales (palabras del idioma pero no correctas en el contexto).
- Errores de estilo (palabras redundantes, ambiguas o repetidas).

De la misma forma, los errores de redacción se pueden clasificar, según su naturaleza, en dos clases: *errores ortográficos*, palabras no presentes en el idioma; y *errores gramaticales*, palabras del idioma usadas incorrectamente en el contexto [1]. Por lo tanto, el objetivo de un corrector ortográfico es señalar al usuario las palabras del texto que se encuentran escritas de manera errónea, y sugerir la palabra más apropiada dentro de una serie de palabras válidas en el idioma. En cambio, la tarea del corrector gramatical es más compleja, ya que se encarga de verificar la correcta construcción de una oración, como la concordancia de género y número, tiempos verbales, etcétera; por lo que no siempre pueden descubrir errores en los que la palabra es ortográficamente correcta pero su uso es incorrecto en un contexto específico (por ejemplo: *"una obra de teatro popular* ***mexicana****"*).

Por otra parte, los errores ortográficos producidos por las personas durante la composición de un texto pueden producirse por alguna de las siguientes razones:

- por equivocaciones al teclear los caracteres, los que son *tipográficos*,
- o por desconocimiento de las reglas de ortografía, los llamados *cognitivos*.

Así, por ejemplo, la palabra *"teirra"* es un error tipográfico causado por transponer los caracteres *"e, i"*. Esto se debió a que se presionaron en orden inverso las teclas correspondientes a los caracteres mencionados. Y un ejemplo en el que aparece un error cognitivo es *"canpo"* cuando la palabra correcta es *"campo"*. La causa de este error es el desconocimiento de la regla ortográfica que establece que antes de *p* o *b* se escribe *m*.

Del mismo modo, los errores de carácter ortográfico ya sean tipográficos o cognitivos se pueden diferenciar, de acuerdo con el error cometido, en las categorías siguientes [1]:

- Error por sustitución de una letra por otra (por ejemplo, *"elije"* en lugar de *"elige"*).
- Error por inserción de una letra extra (por ejemplo, *"dirrección"* en lugar de *"dirección"*. Un caso particular es la separación de palabras cuando se introduce un espacio entre ellas, tal como *"video juegos"* por *"videojuegos"*).

- Error por eliminación de una letra (por ejemplo, *"bibioteca"* en lugar de *"biblioteca"*. De igual manera, un caso particular es la unión de palabras, como se muestra en *"denuevo"* por *"de nuevo"*).
- Error por transposición de dos letras adyacentes (por ejemplo, *"haora"* en lugar de *"ahora"*).

En [2] se definió a los errores simples como las palabras que presentan una sola ocurrencia de uno de los errores anteriores: sustitución de una letra, inserción de una letra, eliminación de una letra o transposición de dos letras. Por consiguiente, se pueden catalogar los errores ortográficos como errores simples o errores múltiples según el número de errores que los diferencia de las palabras correctas, esto es, la cantidad de caracteres erróneos presentes en el error. Por lo tanto, los errores simples presentan una sola transformación y los errores múltiples más de una.

3. Distancia de edición

La medida básica utilizada para calcular la similitud entre palabras es la distancia de edición también conocida como *distancia de Levenshtein*[5]. El algoritmo cuenta la cantidad de operaciones requeridas para convertir una cadena de caracteres en otra, es por esto que, las únicas operaciones de edición permitidas son:

- Inserción de un carácter,
  alumo $->$ *alumno* (agregar *'n'* entre la *'m'* y la *'o'*).
- Eliminación de un carácter,
  transladan $->$ *trasladan* (quitar la primer *'n'*).
- Sustitución de un carácter por otro,
  numero $->$ *número* (reemplazar la *'u'* por *'ú')*.

Notese que la respuesta de este algoritmo es un valor numérico. Por ejemplo, la distancia entre *"tsetal"* y *"tzeltal"* es dos, dado que se necesitan dos operaciones elementales para transformar una palabra en otra:

1. *tsetal* $->$ *tzetal* (sustituir la *'s'* por *'z'*)
2. *tzetal* $->$ *tzeltal* (insertar la *'l'*)

Con el propósito de mejorar dicho algoritmo Damerau agregó una operación más, la transposición de dos caracteres adyacentes [2].

- Transposición de un carácter con otro,
  paelta $->$ *paleta* (permutar *'e'* con *'l'*).

Esta modificación de la distancia clásica de Levenshtein para contar la transposición de caracteres como una sola operación y no como dos distintas, produjo una medida de edición diferente conocida como *distancia Damerau-Levenshtein*. Damerau también encontró que el 80% de todos los errores ortográficos corresponde a palabras erróneas con distancia de edición igual a uno respecto a la palabra que originalmente debía escribirse. Esto significa que el 80% de los errores son de carácter tipográfico y se encuentran en una de las cuatro categorías de error descritas en la sección anterior.

3.1. Modelo de corrección

En general, el método de corrección basado en la distancia de Levenshtein trata de encontrar en el diccionario la palabra correcta c que es más similar a la palabra no reconocida w. En otras palabras, si la palabra es errónea entonces se obtendrán todas sus transformaciones posibles mediante la inserción, eliminación, sustitución y transposición de cada uno de sus caracteres. Después de generar todas las transformaciones de la palabra mal escrita, cada una de ellas se busca en el diccionario y las palabras que se encuentren en él se agregan a la lista de alternativas. La mejor sugerencia de corrección será la de menor distancia a la palabra errónea.

Como ejemplo, en la Tabla 1 se muestran todas las palabras generadas a partir de la palabra errónea *"**vidro**"* con una sola operación de edición. En este caso, las correcciones que se tienen son *"cidro"*, *"video"* y *"vidrio"*. La problemática consiste ahora en definir cuál es la corrección más apropiada, puesto que originalmente todas las operaciones de edición tienen el mismo costo y éste es unitario.

Tabla 1. Posibles transformaciones de la palabra *"vidro"* con distancia de edición uno.

Operación	Palabras generadas
Eliminación	*idro, vdro, viro, vido, vidr*
Inserción	*avidro, bvidro, cvidro, ..., zvidro, ..., vidroa, vidrob, ..., vidroz*
Sustitución	*aidro, bidro,* ***cidro****, ..., zidro, ..., vidra, vidrb, ..., vidrz*
Transposición	*ivdro, vdiro, virdo, vidor*

En estas condiciones, la distancia sólo es un valor numérico que cuenta el número de transformaciones, de modo que, mientras más transformaciones haya mayor será el valor del costo y viceversa. Por supuesto, esta limitación se corrige si se asignan costos distintos a las operaciones [7]. Por ejemplo, en el español mexicano es más probable encontrar la letra 'a', esto justificaría asignar un costo de sustitución menor cuando se cambie un carácter por 'a' que por otro menos frecuente. En la Tabla 2 se muestran las frecuencias de las letras en el vocabulario fundamental del español de México el cual tiene 842 vocablos y es considerado una muestra representativa del español mexicano (y de la lengua española en su conjunto) [4].

Por otra parte, para el idioma español cuyo alfabeto se compone de 27 letras: *a, b, c, d, e, f, g, h, i, j, k, l, m, n, ñ, o, p, q, r, s, t, u, v, w, x, y, z*; una palabra de longitud n produce: n eliminaciones, $n-1$ transposiciones, $26n$ sustituciones y $26(n+1)$ inserciones lo que da un total de $54n+25$ palabras generadas. Mas aún, de este conjunto de palabras solamente un pequeño número serán palabras reales presentes en el diccionario del idioma. En el ejemplo anterior, para la palabra *"vidro"* de longitud 5, se obtuvieron 295 transformaciones y sólo 3 palabras válidas.

Tabla 2. Frecuencias de letras en orden descendente (extraídas de [4]).

Letra	Frec.	Letra	Frec.	Letra	Frec.
a	631	l	193	h	31
r	611	d	191	j	27
e	584	u	180	z	25
o	425	m	174	q	18
i	379	p	173	y	15
n	324	g	78	ñ	11
c	309	b	68	x	11
t	282	v	52	k	0
s	239	f	46	w	0

Por consiguiente, el proceso de generar todas las transformaciones puede ser costoso computacionalmente. No obstante, si se considera la frecuencia de las letras, entonces se podría reducir la cantidad de palabras a evaluar. Por ejemplo, en el caso del español usual mexicano, las letras *'k'* y *'w'* podrían no usarse en las inserciones o sustituciones porque su frecuencia es cero (ver Tabla 2).

3.2. Modelo de lenguaje

En [8] se describe la implementación de un corrector ortográfico cuyo modelo de corrección es el siguiente: dada una palabra errónea, se trata de encontrar la palabra en el diccionario con mayor probabilidad de corregirla. Es decir, dada una palabra w se intenta encontrar la corrección c_i, de entre todas las posibles correcciones, que maximice la probabilidad de corregir a w, esto es:

$$argmax\ P(c_i|w). \tag{1}$$

Que de acuerdo con el *teorema de Bayes*, esto es equivalente a:

$$argmax\ \frac{P(w|c_i)P(c_i)}{P(w)}. \tag{2}$$

Pero, puesto que $P(w)$ es la misma para toda corrección c_i, la ecuación 2 se reduce a:

$$argmax\ P(w|c_i)P(c_i), \tag{3}$$

donde $P(c_i)$ es la probabilidad de que la corrección sugerida c_i ocurra en el idioma utilizado; $P(w|c_i)$ es la probabilidad de que la palabra w haya sido escrita en lugar de c_i; y $argmax$ es la función que determina el valor máximo de la ecuación 3 para encontrar la palabra correcta.

A su vez, la probabilidad $P(c_i)$ se aproxima directamente como la frecuencia de ocurrencia de la palabra c_i en el corpus. Mientras que $P(w|c_i)$ se aproxima como el número de veces que se escribe w en lugar de c_i por uno de los errores ortográficos simples (sustitución, inserción, eliminación o transposición). De modo que, para estimar dicha probabilidad se requiere determinar tanto la frecuencia como el tipo de errores que ocurren en el idioma tratado. Por esta razón, en [8] se optó por definir que las correcciones c_i con distancia de edición uno respecto a w son más probables que las c_i con distancia de edición mayor que uno. Así, de todas las correcciones generadas que aparecen en el diccionario, el sistema elige como correcta aquélla que tenga mayor probabilidad de ocurrencia en el corpus de referencia.

4. Método propuesto

Este trabajo propone una modificación al algoritmo propuesto por Levenshtein [5]. Para ello, se asignan a las operaciones de edición costos basados en la frecuencia de las letras que componen a las palabras. Este esquema de ponderación intenta maximizar la distancia entre w y c_i, el error y la corrección; en particular, cuando esta última es poco probable en el vocabulario del español de México ya que utiliza caracteres menos frecuentes en él.

4.1. Costos de edición

En esta primera propuesta solamente se modificará el costo de las operaciones de inserción y sustitución, de modo que los costos de las operaciones son los siguientes:

- Costo de inserción y sustitución: si el carácter es el más frecuente entonces el costo de la operación es el mínimo, en caso contrario el costo se incrementará.
- Costo de eliminación y transposición: el costo es uno para cualquier carácter.

Con esta modificación se observa que, cuando el costo de edición entre dos palabras es cercana a cero su distancia entre ellas es casi nula produciendo un valor de cercanía alto. Por el contrario, si las palabras no son similares en su grafía, entonces su distancia aumentará.

4.2. Composición del diccionario

Como se ha dicho, cuando se construye la lista de sugerencias para un error ortográfico se necesita comprobar que cada posible corrección esté presente en el idioma, razón por la cual es necesaria la composición de un diccionario del español. Así que, para la composición del diccionario se utilizó la lista de todas las formas ortográficas presentes en el Corpus de Referencia del Español Actual (CREA)[4] que cuenta con casi 140 000 documentos; más de 154 millones de

[4] http://corpus.rae.es/lfrecuencias.html

palabras procedentes de textos de todos los países hispánicos y producidos entre 1975 y 2004; más de 700 000 palabras diferentes y más de 100 materias distintas.

El criterio para decidir si una palabra pertenece o no al idioma español fue que se encontrará en la lista de palabras y que su *frecuencia normalizada* en el corpus fuera mayor a un umbral, el cual fue establecido a 0.5 con base en experimentos; si la palabra no cumple con estas condiciones se considera un error ortográfico.

4.3. Algoritmo

El método de corrección involucra todas las técnicas que fueron descritas anteriormente y para ser más específicos se explica a continuación:

1. Separar cada palabra del texto, estas palabras se evaluarán individualmente.
2. Detectar palabras erróneas mediante el uso del diccionario. Para identificar a las palabras que no están escritas correctamente, se compara cada palabra con las existentes en un diccionario.
3. Para cada error generar todas sus posibles transformaciones (con distancia de edición uno), que a su vez se buscan en el diccionario para eliminar todas aquellas que no estén presentes en el mismo.
4. Ordenar la lista de palabras correctas de acuerdo con la distancia que estas correcciones tengan con la palabra errónea y con la probabilidad de ocurrencia en el corpus.
5. Seleccionar la sugerencia con costo más bajo.

5. Experimentos

Con el fin de evaluar la precisión del método propuesto, para detectar y corregir errores, se utilizaron 55 oraciones cada una con un error ortográfico, dando como resultado 45 errores distintos. La fuente de este material fue la Fe de erratas de los libros de educación primaria del ciclo escolar 2013-2014 publicada por la Secretaría de Educación Pública (SEP) [3].

Aunque en realidad fueron 117 los errores ortográficos, gramaticales y semánticos los que se detectaron en los libros (de todos los niveles formativos y en la totalidad de sus asignaturas) sólo se utilizaron los errores ortográficos simples. De ahí que, errores ortográficos como *"**tsetal**"* o *"**Iztacihuátl**"* no se consideraron en las pruebas porque la corrección, *"tzeltal"*, del primer error se obtiene con 2 operaciones (una sustitución y una inserción) y la corrección, *"Iztaccíhuatl"*, para el segundo error requiere de 3 operaciones (una inserción y dos sustituciones). La Tabla 3 muestra algunas de las oraciones con problemas de ortografía utilizadas para probar el método.

Además, se implementaron los siguientes métodos de corrección:

– Método base: usa la distancia de edición clásica donde cada operación de edición tiene un costo uniforme de 1. En este modelo se selecciona la palabra más frecuente en el corpus, como se realizó en [8].

Tabla 3. Ejemplos de errores ortográficos reales.

Dice	**Debe decir**
"Relaciones entre los ***numeros****"*	*"... números"*
*"****Rescribir*** *canciones conservando la rima"*	*"Reescribir ..."*
"A la ***vibora*** *de la mar"*	*"... víbora ..."*
"En esta lección ***alaborarás*** *un ritmo visual"*	*"... elaborarás ..."*
*"****Elije*** *alguno que te haya gustado e imita su postura"*	*"Elige ..."*
*"****Codice*** *florentino, siglo XVI"*	*"Códice ..."*
"Si Juanito rompió el ***vidro*** *a propósito"*	*"... vidrio ..."*
"Juan ***Nepomuseno*** *Almonte"*	*"... Nepomuceno ..."*
*"****Gadalajara****, Jalisco"*	*"Guadalajara ..."*
"Mi único medio de ***trasporte*** *era un burro"*	*"... transporte ..."*

– Método propuesto: usa la modificación a la distancia de edición original para que las operaciones tengan un costo diferente y luego se selecciona la palabra más utilizada en el corpus.

5.1. Resultados

A continuación se presentan los resultados del algoritmo propuesto en comparación con los correctores ortográficos de Google Docs[5] y Microsoft Word 2016[6]. En la Tabla 4 se resume la evaluación, de la detección y corrección de errores, realizada con dichos correctores y el método propuesto. De los resultados obtenidos se puede observar, por una parte, que los métodos basados en la distancia de Levenshtein detectaron el 86.6% de los errores posibles; notese que ambos obtienen el mismo resultado porque utilizan el mismo diccionario. Por otra parte, el método propuesto corrigió el 71.1% de errores y el método base corrigió el 73.3%, esto significa que los algoritmos son comparables en funcionamiento. Por último, Google Docs identificó sólo el 44.4% de errores y Word 2016 detectó el 77.7%; aunque no se conocen los algoritmos que utilizan estas herramientas, es muy probable que utilicen métodos más complejos, a pesar de ello, sólo corrigieron de forma apropiada el 40% y 66.6% de errores, respectivamente.

Conviene subrayar que los resultados se evaluaron como apropiados cuando la primera sugerencia para el error coincidió exactamente con la corrección establecida. Por lo tanto, los resultados del algoritmo propuesto podrían mejorar si se toman en cuenta todas las sugerencias encontradas para los errores. Por ejemplo, al revisar los experimentos se observó que la lista de sugerencias para el error *"****rescribir****"* fue *"describir"*, *"reescribir"*, *"prescribir"*, *"escribir"* y para *"****vidro****"* fue *"video"*, *"vidrio"*, *"vitro"*, *"cidro"*.

[5] https://docs.google.com
[6] https://www.microsoft.com

En general, se observa que el algoritmo que se propone permite identificar y corregir adecuadamente los errores de tipo ortográfico. Sin embargo, debido a su simplicidad, el corrector falla en detectar el error cuando este produce una palabra válida, distinta de la que el usuario deseaba escribir. Así, por ejemplo, la primer sugerencia para el error *"**elije**"* fue *"elija"*.

Finalmente, de la lista completa de palabras analizadas, que se muestran en la Tabla 5 del Anexo 1, se puede observar que ninguna de las herramientas de corrección identifica a todas las palabras erróneas aunque estas sean comunes, por ejemplo: no detectan a *"**closet**"* ni a *"**arboles**"* por lo que se puede concluir que esas palabras sí se encuentran en el diccionario y además, tienen una frecuencia de ocurrencia alta en el corpus de referencia. Así mismo, se observa que el error *"**físicomotrices**"* fue detectado pero no corregido, esto sucedió porque la palabra generada *"fisicomotrices"* aunque es correcta no se encontró en el diccionario utilizado.

Tabla 4. Resultados de la evaluación de los correctores con los 45 errores.

	Errores			
Método	**Detectados**	**No detectados**	**Corregidos**	**No corregidos**
Base	39	6	33	12
Propuesto	39	6	32	13
Google Docs	20	25	18	27
Word 2016	35	10	30	15

6. Conclusiones

El aporte principal de este trabajo es la utilización de la frecuencia de las letras para desarrollar un método de corrección ortográfica orientado al español de México. Este enfoque requiere poca intervención humana puesto que solo necesita de un corpus para construir el diccionario y el modelo de lenguaje. Por supuesto que, el corrector ortográfico como casi todas las herramientas que se construyen para procesar textos depende, entre otras cosas, del dominio de aplicación y del idioma a tratar.

Indiscutiblemente, este método no utiliza ninguna de las propiedades lingüísticas de la palabra ni el contexto en que ésta se utiliza. Esta característica impide la corrección de ciertos errores, en concreto, cuando la palabra sugerida es correcta pero no se encuentra en el diccionario utilizado; en estos casos la palabra se seguirá considerando errónea. No obstante, los resultados obtenidos en la evaluación son prometedores puesto que se trata de un corrector ortográfico de propósito general.

El trabajo futuro consistirá en elaborar un corpus de errores reales, ya que de él se podrán obtener los casos de error más frecuentes y en consecuencia un

modelo de lenguaje que proporcionará información para estimar las probabilidades de las sugerencias, esto con el fin de mejorar la precisión al elegir la palabra correcta. De igual manera, las cuestiones relacionadas con los errores ortográficos múltiples, gramaticales y de estilo, se tratarán en trabajos posteriores.

Agradecimientos. Agradezco las observaciones y sugerencias de los revisores, que sin duda alguna contribuyeron a mejorar la presentación y calidad de este trabajo.

Referencias

1. Castro, D.: Métodos para la corrección ortográfica automática del español, Tesis de Maestría, Facultad de Matemáticas y Computación, Universidad de Oriente, Santiago de Cuba (2012)
2. Damerau, F.: A technique for computer detection and correction of spelling errors, Communications of the ACM, vol. 7(63), pp. 171–176 (1964)
3. Fe de erratas de los libros de educación primaria del ciclo escolar 2013-2014, Secretaría de Educación Pública (SEP) (2013)
4. Lara, L. F.: Diccionario del Español de México (DEM), El Colegio de México, A.C., Consultado el 4 de Septiembre del 2016 en: http://dem.colmex.mx
5. Levenshtein, V.: Binary codes capable of correcting deletions, insertions, and reversals, Soviet Physics Doklady, vol. 10(8), pp. 707–710 (1966)
6. Naber, D.: A rule-based style and grammar checker (2003)
7. Needleman, S., Wunsch, C.: A general method applicable to the search for similarities in the amino acid sequence of two proteins, Journal of molecular biology (Elsevier), vol. 48 (3), pp. 443–453 (1970)
8. Norvig, P.: How to write a spelling corrector (2007), Consultado el 4 de Septiembre del 2016 en: http://norvig.com/spell-correct.html

Anexo 1: Resultados experimentales en detalle

En este anexo se presentan los resultados que comparan la exactitud del método propuesto con la de otras herramientas de corrección (ver Tabla 5). La primera columna es la palabra errónea, la segunda es la corrección apropiada y de la tercera a la sexta columna están los resultados de la corrección. En dicha tabla, el símbolo '−' corresponde a los errores no detectados, '+' señala los errores detectados y corregidos, '×' señala los errores detectados pero no corregidos y un '∘' indica los errores detectados que no tienen sugerencias, por lo cual, tampoco fueron corregidos.

Tabla 5. Resultados para todas las palabras erróneas con diferentes correctores.

Error	**Corrección**	**Método Base**	**Método Propuesto**	**Google Docs**	**Word 2016**
numeros	números	+	+	−	+
panques	panqués	×	×	−	−
escola	escolar	−	−	−	×
exijen	exigen	+	+	+	+
rescribir	reescribir	×	×	×	−
vibora	víbora	+	+	−	+
prrault	perrault	+	+	+	×
atribuída	atribuida	×	+	−	+
heróicos	heroicos	+	+	−	+
quetzalcoatl	quetzalcóatl	+	+	−	+
nauseas	náuseas	+	+	−	−
boiler	bóiler	×	×	−	−
podium	pódium	+	+	−	+
sonreirle	sonreírle	+	+	−	+
closet	clóset	−	−	−	−
alberges	albergues	+	+	+	−
kenedy	kennedy	+	+	−	+
compañia	compañía	+	+	−	+
ditrosionan	distorsionan	+	+	+	+
ocaciona	ocasiona	+	+	+	+
contrarestan	contrarrestan	+	+	+	+
alaborarás	elaborarás	×	×	−	+
elije	elige	+	×	+	−
leccion	lección	+	+	−	+
iconográgico	iconográfico	+	+	−	+
bibioteca	biblioteca	+	+	+	+
ademas	además	−	−	−	+
alumo	alumno	+	+	−	×
físicomotrices	fisicomotrices	∘	∘	−	×
provacar	provocar	+	+	+	+
mantenela	mantenerla	+	+	+	+
vidro	vidrio	+	×	×	+
codice	códice	+	+	−	+
crea-tividad	creatividad	+	+	+	×
gobieno	gobierno	+	+	+	+
trasporte	transporte	−	−	+	−
sequias	sequías	+	+	×	−
arboles	árboles	−	−	−	−
gadalajara	guadalajara	+	+	+	+
simboligía	simbología	+	+	+	+
ligüísticos	lingüísticos	+	+	−	+
transladan	trasladan	+	+	+	+
terrirorio	territorio	+	+	+	+
cristobal	cristóbal	−	−	−	+
nepomuseno	nepomuceno	+	+	+	+

ISSN 1870-4069

Análisis automático de conversaciones para determinar el comportamiento de pederastas

Beatriz Beltrán, Darnes Vilariño, David Pinto

Benemérita Universidad Autónoma de Puebla,
Facultad de Ciencias de la Computación, México

{bbeltran, darnes, dpinto}@cs.buap.xm

Resumen. Debido a que, en la actualidad, el acceso a medios masivos se encuentra al alcance de cualquier persona, en especial de los niños y sobre todo al no existir restricciones para su acceso, personas sin escrúpulos, tales como los pederastas y pedófilos andan en búsqueda de nuevas víctimas en estos sitios. Es por eso que encontrar formas de identificar acciones que llevan a cabo los pederastas, tales como, propuestas indebidas toma relevancia. En el presente trabajo, se realiza una investigación acerca de la forma en que los pedófilos convencen a sus posibles víctimas, tal análisis se realiza a partir de las intervenciones que tienen los pederastas en salas de conversaciones.

Palabras clave: Pederastas, víctimas, reconocimiento de patrones.

Automatic Analysis of Conversations to Determine Pederast Behaviors

Abstract. Nowadays, access to mass media is available to everyone, especially to children, there exist an absence of restrictions for accessing those media, therefore, unscrupulous persons such as pedophiles and pederasts take advantage of this issue for looking new victims in these sites. This is why finding ways to identify actions carried out by pederasts, such as, inappropriate proposals becomes to be relevant. In this paper we present an investigation about the manner in which pederasts convince their potential victims; such an analysis is performed by using conversations established by pederasts in chat rooms.

Keywords. Pederasts, victims, pattern recognition.

1. Introducción

Se han desarrollado diversas investigaciones para la búsqueda de depredadores sexuales en la red. Con el desarrollo de internet cada día es más fácil que un individuo

trate de interactuar con otros en búsqueda de algún favor sexual, sin que la víctima lo detecte de manera directa.

La mayoría de los trabajos que abordan esta temática toman como punto de referencia la investigación presentada en Pendar [1], donde se realiza un estudio piloto sobre el uso de técnicas de clasificación automática de textos para identificar depredadores sexuales en línea. Se trabaja con un corpus obtenido del sitio Perverted Justice[1], el cual recopila conversaciones de pedófilos con personas que se hacen pasar por niños o adolescentes en la región de Estados Unidos. Este estudio se presenta como una investigación inicial para la detección del *grooming attack*, definido en Harms [2] como "proceso de comunicación por el cual un autor aplica estrategias de búsqueda de afinidad, mientras que simultáneamente adquiere información sobre sus víctimas con el fin de desarrollar las relaciones que resulten en cumplimiento de su necesidad, por ejemplo, acoso sexual físico".

En los experimentos realizados, se identifican los textos de los depredadores sexuales de las víctimas (clasificación en dos clases), por lo tanto, se separan los diálogos por autor, posteriormente se extraen las palabras cerradas, además de hacer una corrección a las conversaciones, como quitar las letras repetidas de las palabras, remover signos de puntuación, emoticones, entre otros y se obtienen unigramas, bigramas y trigramas de palabras. Los experimentos fueron realizados utilizando SVM y k-NN. Se hicieron varias pruebas con diferentes características (de 5,000 a 10,000), obteniendo los mejores resultados con un conjunto de 10,000 características con k-NN (k=30), con un *F-measure* de 94%.

En los experimentos realizados en Michalopoulos & Mavridis [3] también se intenta reconocer el *grooming attack* utilizando técnicas de clasificación de documentos para la creación de patrones. Se comparan siete diferentes algoritmos para un corpus que contiene tres clases:

1. Ganarse a la víctima.
2. Cultivo de una relación amorosa.
3. Petición de favor sexual o abuso.

Estas clases describen el nivel de relación existente entre el depredador y la víctima, y permiten hacer una clasificación de los tipos de depredadores. Se utilizó Naïve Bayes, el algoritmo "Esperanza-Maximización" (EM), k-NN (con k=24), EM-SIMPLE, TF-IDF, máxima entropía y SVM, obteniendo una mejor precisión con Naïve Bayes (0.96), además de un tiempo menor de clasificación.

En Miah et al. [4] se identifica cuando ocurre explotación infantil en una conversación. Realizan una comparación entre el uso de características basadas en términos con las extraídas utilizando la herramienta *Linguistic Inquiry and WordCount* (LIWC) [5], el cual es un sistema para calcular el grado en que las personas usan ciertas categorías de palabras para analizar aspectos como decepción, honestidad, entre otros. LIWC acepta un texto de entrada y produce variables de salida, las cuales se agrupan en 4 categorías:

[1] http://www.perverted-justice.com/

1. **Procesos lingüísticos:** Palabras, pronombres, verbos, entre otros.
2. **Procesos psicológicos:** Sociales, afectivos, cognitivos, de percepción, biológicos, entre otros.
3. **Preocupaciones personales:** Trabajo, casa, dinero, religión, etc.
4. **Categorías del habla:** Asentimientos, influencias, rellenos.

Las conversaciones fueron obtenidas de los sitios perverted.justice.com, chatdump.com y www.fugly.com, y fueron etiquetadas de acuerdo a las siguientes categorías:

- EI: Explotación infantil (200 chats).
- FS: Fantasías sexuales entre adultos (85 chats).
- CG: Chat general sin contenido sexual (107 chats).

Se construyeron 4 distintos conjuntos de datos con las diferentes categorías (EIFS, EI-CG, FS-CG y EI-FS-CG). Estos se clasificaron con Naïve Bayes, árboles de decisión y clasificación vía regresión. Se reporta el número de conversaciones clasificadas correctamente en cada caso, obteniendo los mejores resultados con el uso de LIWC con Naïve Bayes, particularmente en la detección de la categoría "EI", con alrededor de 185 conversaciones detectadas en los 4 conjuntos construidos.

La presente investigación intenta encontrar patrones de escritura en las conversaciones entre los depredadores y sus víctimas, partiendo de la hipótesis de que cuando la longitud de una de las intervenciones (ya sea depredador o víctima) es más larga que la otra, entonces, asumimos que se está aportando más información y, por tanto, esa fase podría ser de importancia en la detección de acoso sexual, por ejemplo.

El trabajo está dividido como sigue. En la Sección 2, se describe la metodología que se llevó a cabo para el análisis de la hipótesis planteada. En la sección 3 se analizan los resultados, partiendo de la metodología sobre un análsis de las conversaciones. Finalmente, en la sección 4 se presentan las conclusiones del presente trabajo.

2. Metodología

Se trabajó con un corpus obtenido del sitio Perverted Justice, de donde se extrajeron conversaciones de pederastas. Perverted Justice tiene como objetivo el prevenir a las familias del acoso sexual de pederastas, debido a que en redes sociales y salas de chat es muy simple para estos individuos conectarse con mujeres y niños; estas personas por lo regular se hacen pasar por menores para iniciar una plática, hasta lograr su objetivo. El sitio mantiene información actualizada de las personas que fueron capturadas.

En este trabajo se propone detectar patrones de comportamiento en las conversaciones que mantienen los pederastas con una víctima. El procedimiento propuesto se describe en la Fig.1; se inicia con la descarga de conversaciones desde el sitio web Perverted Justice, de donde se obtiene un archivo por cada pederasta. Posteriormente se procede a obtener el grado de interés dentro de la plática, obteniendo una gráfica para cada conversación de cada uno de los pederastas, para finalmente realizar un análisis basado en grado de interés en la conversación.

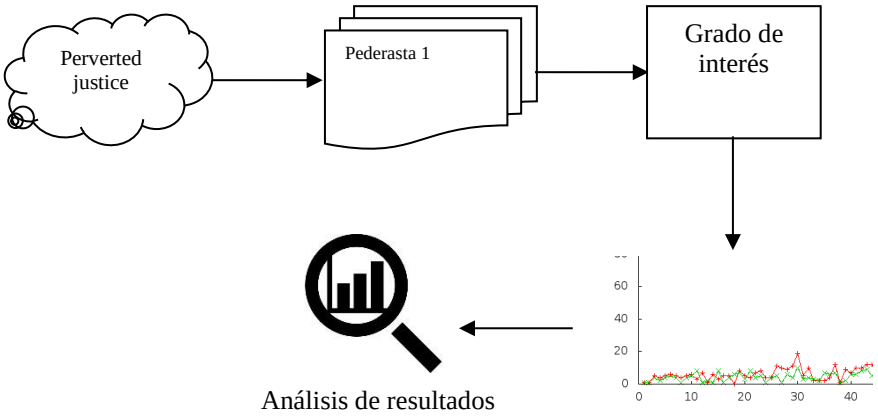

Fig. 1. Esquema del Sistema en general.

En total se descargaron 20 conversaciones correspondientes a 20 diferentes depredadores sexuales. Las características de las conversaciones se presentan en la Tabla 1.

Tabla 1. Características de las 20 conversaciones.

	Total de Intervenciones	**Longitud Promedio**	**Tamaño Vocabulario**	**Cantidad de Palabras**
Depredador 1	360	9.744	529	3836
Victima 1	360	4.369	299	1657
Depredador 2	101	1.762	26	155
Victima 2	101	5.425	153	540
Depredador 3	279	9.358	405	2665
Victima 3	279	6.602	333	1901
Depredador 4	321	15.336	619	4957
Victima 4	321	7.579	394	2450
Depredador 5	142	13	209	1856
Victima 5	142	7.352	214	1044
Depredador 6	773	10.126	897	8749
Victima 6	773	6.598	653	5388
Depredador 7	770	5.906	711	4598
Victima 7	770	5.210	671	4063
Depredador 8	4921	7.011	3679	35164
Victima 8	4921	5.036	2662	24876
Depredador 9	68	6.235	63	568
Victima 9	68	7.897	78	559
Depredador 10	139	6.510	207	1525
Victima 10	139	7.676	218	1130
Depredador 11	367	8.678	545	3191
Victima 11	367	5.896	412	2169

Depredador 12	2258	10.680	2829	28216
Victima 12	2258	4.604	1348	22852
Depredador 13	70	7.471	117	526
Victima 13	70	5.728	110	401
Depredador 14	2146	7.461	2292	16210
Victima 14	2146	5.486	1602	12066
Depredador 15	1296	7.158	1112	8126
Victima 15	1296	6.968	1103	8517
Depredador 16	191	8.005	321	2023
Victima 16	191	4.340	216	997
Depredador 17	394	6.398	410	2741
Victima 17	394	5.111	373	2096
Depredador 18	735	9.997	877	7722
Victima 18	735	5.382	570	4045
Depredador 19	1362	7.253	1264	9880
Victima 19	1362	7.718	1405	10550
Depredador 20	157	5.095	224	1009
Victima 20	157	5.859	247	977

3. Análisis de resultados

Para la búsqueda automática de los patrones y poder detectar el momento en que el acosador inicia el ataque se realizó lo siguiente:

1. Cuantificar el tamaño de cada intervención.
2. Graficar el tamaño de las intervenciones.
3. Analizar las gráficas obtenidas.

A continuación, se realiza un análisis de 2 conversaciones a partir de las gráficas obtenidas, considerando que se detectan los momentos donde se incrementa la interacción entre el acosador y la víctima, en aras de comprobar la hipótesis planteada.

En la Fig. 2, se puede observar que en el período de intervenciones de la 75 a la 100, la víctima tiene más intervenciones que el pederasta, y dentro de la conversación se tiene que durante ese tiempo es cuando la víctima ofrece información personal, como por ejemplo, que actividades lleva a cabo de manera cotidiana, si su mamá se encuentra trabajando, a qué hora puede salir de su casa, etc.

Así mismo, en el intervalo de 125 al 250, es cuando el pederasta tiene una mayor intervención, durante este tiempo es en el cual el pederasta inicia el convencimiento para llegar a tener un contacto físico, realizando comentarios de índole sexual.

Parte de la conversación analizada se muestra en la Fig. 3, cabe hacer mención que las conversaciones son realizadas en el idioma inglés.

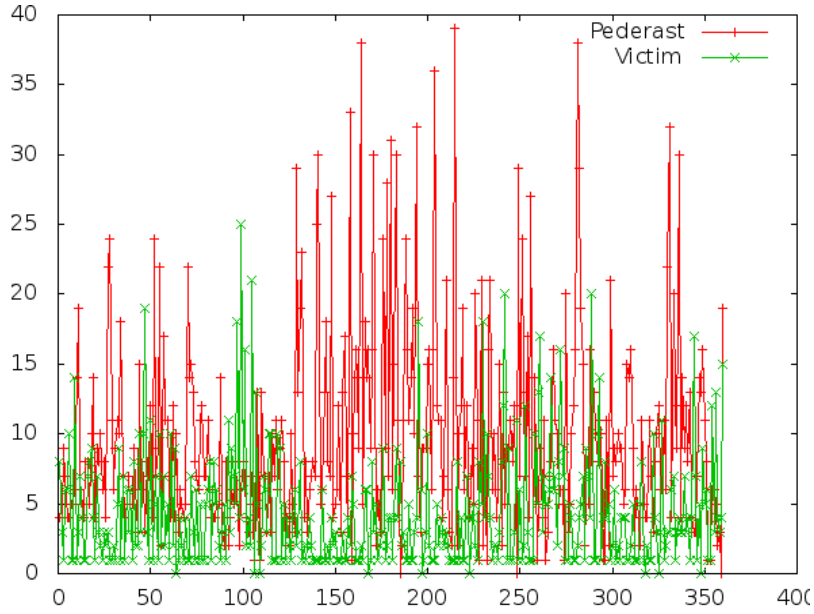

Fig. 2. Conversación del pederasta con el sobrenombre=*Arthinice*.

189 arthinice (6:52:47 PM): know how it has that silky slit in the middle?
189 arthinice (6:53:31 PM): the part between the pussy lips? (A description. Great)
189 sadlilgrrl (6:53:31 PM): yeah
190 arthinice (6:53:48 PM): i'd want to stroke my tongue up and down that part (I bet you would)
190 sadlilgrrl (6:54:00 PM): oh
190 sadlilgrrl (6:54:03 PM): does that feel good?
191 arthinice (6:54:05 PM): especially if it is really wet there
191 arthinice (6:54:16 PM): feel good to me? or to you?
191 sadlilgrrl (6:54:29 PM): either?
192 arthinice (6:54:37 PM): i like the feel and the taste
192 arthinice (6:54:47 PM): but it is you that would be feeling good
192 arthinice (6:54:54 PM): that tingly feeling
192 sadlilgrrl (6:55:04 PM): cool
193 arthinice (6:55:04 PM): but a lot lot lot stronger
193 arthinice (6:55:47 PM): think you'd like that?
193 sadlilgrrl (6:56:28 PM): i think so

Fig. 3. Conversación analizada.

En otra conversación analizada, se puede ver un comportamiento muy similar, en las primeras intervenciones de la conversación, el pederasta se concreta a preguntar cosas acerca de la vida cotidiana de la víctima, y es a partir de la intervención 100, cuando inicia el ataque con comentarios de posesión y en donde en base a una pregunta de la

víctima, acerca de un sitio, que se hace llamar "*Legion of master*", inicia una conversación para controlar a su víctima a tal grado que lo llama "*master*".

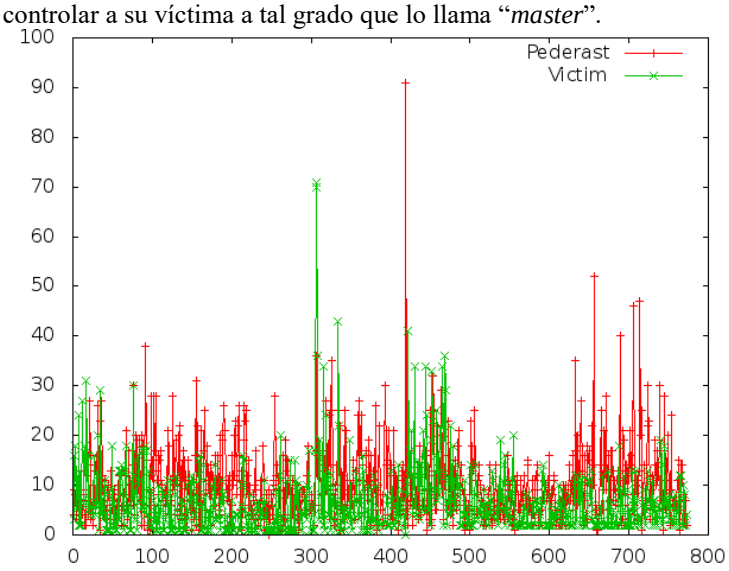

Fig. 4. Conversación del pederasta con el sobrenombre=*Fleet_captain_jaime_wolfe*.

En el punto 306, la víctima tiene que disculparse por lo que el pederasta llama una ofensa y por lo cual la víctima se ve en la necesidad de pedir perdón además de explicar situaciones familiares.

617 fleet_captain_jaime_wolfe (6:02:35 PM): Hello girl... How does my girl feel? (He's very possessive.)
617 sadlilgrrl (6:02:39 PM): much better master
618 fleet_captain_jaime_wolfe (6:02:51 PM): Good... And, is she looking forward to tonight?
618 sadlilgrrl (6:02:59 PM): very much so master
619 fleet_captain_jaime_wolfe (6:03:09 PM): Good... Does she need what will happen?
619 sadlilgrrl (6:03:15 PM): yes master
620 fleet_captain_jaime_wolfe (6:03:28 PM): Good, & do you have anything special you want me to do to you?
620 sadlilgrrl (6:03:42 PM): whatever you want master
620 sadlilgrrl (6:03:47 PM): im really excited master
621 fleet_captain_jaime_wolfe (6:03:51 PM): Good... So you are up for anything?
621 sadlilgrrl (6:03:56 PM): yes master
622 fleet_captain_jaime_wolfe (6:04:00 PM): How excited? Are you nice & wet for me?
622 sadlilgrrl (6:04:06 PM): yes master
623 fleet_captain_jaime_wolfe (6:04:26 PM): Good... And, are you looking forward to feeling your nipples clamped?

Fig. 5. Conversación analizada.

Al final de la conversación, el control ya llega a tal grado que la víctima solo contesta con yes/no master. En todo momento la conversación gira alrededor del tema sexual y sobre el control que él tiene sobre ella. Además de que demuestra complacencia por la actitud que la víctima toma, al comportarse en todo momento de manera sumisa y con actitud de querer complacerlo en todo.

4. Conclusiones

El problema del acoso a menores en redes sociales ha aumentado cada año, por lo cual el poder identificar patrones que se encuentren asociados a los pederastas, ayudaría a los diferentes elementos de seguridad y a la sociedad en sí misma.

Este trabajo contribuye con el análisis automático de comportamientos para encontrar dichos patrones, y es así como se pudo comprobar que en los momentos en que el acosador requiere información personal de su víctima, se incrementan las intervenciones de la víctima, incluso, cuando ya tiene dominada a la víctima y el pederasta no le gusta lo expresado por su víctima y entonces, esta se tiene que disculpar, se ve un comportamiento semejante de mayor participación de la víctima.

Sin embargo, este comportamiento se invierte completamente cuando el pederasta está realizando la labor de consentimiento, convencimiento e incluso cuando logra algo, como el obtener fotografías, sometimiento o hasta llegar a una cita.

Referencias

1. Pendar, N.: Toward spotting the pedophile telling victim from predator in text chats. In: Proceedings of the International Conference on Semantic Computing, ICSC '07, IEEE Computer Society, 235–241 (2007)
2. Harms, C. M. Grooming: An operational definition and coding scheme. 8(1), 1–6 (2007).
3. Michalopoulos, D. & Mavridis, I.: Utilizing document classification for grooming attack recognition. In: Proceedings of the 2011 IEEE Symposium on Computers and Communications, ISCC '11, IEEE Computer Society, 864–869 (2011)
4. Miah, M. W. R., Yearwood, J., Kulkarni, S.: Detection of child exploiting chats from a mixed chat dataset as a text classification task. In: Proceedings of the Australasian Language Technology Association Workshop 2011, 157–165 (2011)
5. Pennebaker, J. W., Chung, C. K., Ireland, M., Gonzales, A., Booth, R. J.: The Development and Psychometric Properties of LIWC2007. 3–14 (2007)

ISSN 1870-4069

Minería de datos centrada en el usuario para el análisis de la supervivencia y mortalidad de casos de cáncer de mama en mujeres de origen mexicano

Aldair Antonio-Aquino[1], Guillermo Molero-Castillo[2], Rafael Rojano-Cáceres[3], Alejandro Velázquez-Mena[4]

[1] Universidad Veracruzana, MSICU, Facultad de Estadística e Informática, México

[2] CONACYT, México

[3] Universidad Veracruzana, Facultad de Estadística e Informática, México

[4] Universidad Nacional Autónoma de México, Facultad de Ingeniería, México

aquinoaldair@hotmail.com, ggmoleroca@conacyt.mx, rrojano@uv.mx, mena@fi-b.unam.mx

Resumen. Actualmente existen diversos procesos que conducen el desarrollo de un proyecto de minería de datos. Sin embargo, éstos, a pesar de su variedad, no están centrados en el usuario; trayendo como consecuencia aplicaciones con limitaciones de usabilidad y accesibilidad. En este trabajo se presenta una minería de datos centrada en el usuario con base en los fundamentos de la norma ISO 9241-210:2010. Se analizó la supervivencia y mortalidad de casos de cáncer de mama en mujeres de origen mexicano. Los datos utilizados corresponden a registros clínicos del Instituto Nacional del Cáncer de los Estados Unidos. Como resultado se obtuvo una precisión del 87.4%. Además, las pruebas de usabilidad basado en heurísticas permitieron identificar mejoras de interacción entre la aplicación y los usuarios finales.

Palabras clave: Cáncer de mama, DCU, minería de datos, origen mexicano.

User-Centered Data Mining for the Analysis of Survival and Mortality of Breast Cancer Cases in Woman of Mexican Origin

Abstract. Currently, there are several processes that lead the development of a data mining project. However, these, despite their variety, are not user-centered; consequently, there are applications with limitations of usability and accessibility. This paper presents a data mining user-centered based on the fundamentals of ISO

9241-210:2010. Survival and mortality of breast cancer cases in women of Mexican origin were analyzed. The data used correspond to clinical records of the National Cancer Institute of the United States. As a result, 87.4% accuracy was obtained. In addition, usability testing based on heuristics helped to identify improvements for interaction between the application and end-users.

Keywords. Breast cancer, data mining, Mexican origin, UCD.

1. Introducción

En la actualidad, se realizan diversas actividades físicas, domésticas, laborales, de investigación, entre otras, que implican un proceso de creación y almacenamiento de nueva información. Esta información puede llegar a ser valiosa para el proceso de la toma de decisiones, esto si se le aplica los métodos y herramientas adecuadas. Una de estas herramientas es la minería de datos como un proceso para la exploración y análisis de volúmenes de datos para el descubrimiento de conocimiento útil. En este sentido, en la actualidad la minería de datos juega un papel importante en el crecimiento y éxito de las organizaciones; propiciando así la búsqueda de nuevo conocimiento manifestado en patrones de datos, tendencias, reglas, grupos, clasificaciones, entre otros.

Precisamente, como resultado del surgimiento de la minería de datos, es necesario contar con procesos que permitan planificar y guiar el desarrollo de los proyectos. Así, en la actualidad existen diversos procesos con enfoques y funcionamiento distintos. Sin embargo, estos procesos, a pesar de su amplia variedad, no están centrados en el usuario, limitando así la participación de éstos en cada una de las etapas que comprenden; trayendo como consecuencia desarrollos de minería de datos con limitaciones de usabilidad y accesibilidad [1, 2]. Por lo que, existe la necesidad de buscar nuevas formas de analizar y procesar las fuentes de datos. Precisamente, una de estas formas es a través de una minería de datos centrada en el usuario.

La importancia que tiene el usuario en proyectos de descubrimiento de conocimiento es fundamental debido a que éstos poseen diversos conocimientos, estilos cognitivos y otras habilidades mentales que mediante un proceso de análisis se puede lograr un mejor entendimiento de las necesidades, tareas y características que se requieren considerar en el proyecto [1, 3]. Asimismo, la importancia del usuario en la minería de datos no solo está en la exploración de volúmenes de datos para el descubrimiento de conocimiento, sino también en el proceso de la toma de decisiones mediante el uso de herramientas interactivas que sean fáciles de usar, aprender y recordar [4]. Además, para involucrar al usuario como parte importante en el proceso de minería de datos se debe considerar características como: a) privado, para preservar la privacidad de los usuarios; b) personal, para asegurar de que el usuario se beneficie del conocimiento encontrado; c) portátil, para asegurar de que el flujo de datos esté en todas partes y en cualquier lugar; y d) potente, para proporcionar recursos suficientes a los usuarios para el descubrimiento de conocimiento a través de los datos [5].

En este sentido, al ser los usuarios parte esencial en este tipo de proyectos, es crucial involucrarlos desde el análisis y entendimiento del proyecto hasta la validación y presentación de los resultados. Es importante señalar que una interpretación incorrecta

de las necesidades y requerimientos de los usuarios pudiera conducir al fracaso de los proyectos de minería de datos o limitar las expectativas de los usuarios [6].

En este trabajo se presentan los resultados de la investigación sobre la contribución de una minería de datos centrada en el usuario con base en los principios de la norma ISO 9241-210:2010, enfocado al análisis de la supervivencia y mortalidad de casos de cáncer de mama en mujeres de origen mexicano. La fuente de datos utilizada corresponde a registros de la base de datos del Programa de Vigilancia, Epidemiología y Resultados Finales (SEER) del Instituto Nacional del Cáncer de los Estados Unidos.

2. Procesos de minería de datos

A pesar de la amplia variedad de tareas y técnicas de minería de datos es necesario trabajar con un marco de trabajo que permita planear y guiar el desarrollo de los proyectos. Actualmente, uno de los más conocidos es el proceso de descubrimiento de conocimiento en base de datos (Knowledge Discovery in Databases o KDD), el cual consta de una serie de etapas para la generación de conocimiento y la toma de decisiones. Este proceso tiene la característica de ser iterativo e interactivo, y que se orienta a las decisiones que toma el usuario [7], sin embargo, no describe las tareas y actividades específicas que se deben realizar en cada una de sus etapas [8]. Otro de los procesos es SEMMA (Sample, Explore, Modify, Model, Assess), creado por SAS Institute (Statistical Analysis Systems), que lo define como la selección, exploración y modelado de grandes volúmenes de datos para descubrir patrones de interés [9]. Particularmente, SEMMA inicia con un análisis exploratorio de datos, ignorando el análisis y entendimiento del proyecto [10, 11]. Por otra parte, SEMMA está relacionada particularmente al uso de productos comerciales de SAS Institute.

Tabla 1. Principales características de los procesos de minería de datos.

	KDD	**CRISP-DM**	**SEMMA**	**Catalyst**	**Six Sigma**
Fases	– Integración y recopilación – Selección, limpieza y transformación – Minería de datos – Evaluación e interpretación – Difusión y uso	– Entendimiento del negocio – Entendimiento de los datos – Preparación de los datos – Modelado – Evaluación – Despliegue	– Muestreo – Exploración – Modificación – Modelado – Evaluación	– Preparación de los datos – Modelado – Refinar el modelo – Implementar el modelo – Comunicación de resultados	– Definición – Medición – Análisis – Mejora – Control
Etapas iterativas	Si	Si	No	Si	No
Elección de herramientas	Libres y comerciales	Libres y comerciales	Comerciales	Libres y comerciales	Libres y comerciales
Evaluación del resultado	Basado en los objetivos del proyecto	Basado en el modelo y los objetivos del proyecto	Basado en el modelo	Basado en los objetivos del proyecto	Basado en el modelo
Orientada a MD	Si	Si	Si	Si	No
Año	1996	1999	1998	2003	1986

CRISP-DM (Cross Industry Standard Process for Data Mining) es otro proceso utilizado en la actualidad en proyectos de minería de datos [8, 13]. Este se caracteriza por dividir el proyecto en diferentes fases, tareas y actividades [14]. Otro proceso es Catalyst [8] conocido como P3TQ (Product, Place, Price, Time, Quantity) conformado por dos modelos [15, 16]: a) negocio (MII) y b) explotación de información (MIII). MII ofrece una guía para el desarrollo de un modelo de un problema u oportunidad de negocio y MIII proporciona una guía para la realización y ejecución de modelos de minería de datos [15, 16].

Por otra parte, un proceso industrial adaptado a la minera de datos es Six Sigma, definido como un método organizado y sistemático para la mejora de procesos, nuevos productos y servicios basados en métodos estadísticos y científicos con el fin de reducir las tasas de defectos establecidos por el cliente [17]. Six Sigma involucra el análisis de datos, a través de herramientas estadísticas, con el fin de reducir la variación mediante la mejora continua [18]. En la Tabla 1 se presenta un resumen de las principales características de los procesos presentados, los cuales en las últimas décadas han tenido un aumento importante, todos con el propósito de cumplir con los objetivos y requerimientos definidos en los proyectos.

A pesar de que estos procesos cumplen con el objetivo principal de guiar el descubrimiento de patrones de interés en volúmenes de datos, aún carecen de aspectos importantes como una mayor participación del usuario en cada una de las etapas y la presentación eficiente de los patrones de datos obtenidos. Ambos aspectos son fundamentales para una mejor explicación y entendimiento en la generación del nuevo conocimiento. En este mismo sentido, surge la necesidad natural de hacer una minería de datos centrada en el usuario con el propósito de mejorar la experiencia de los usuarios en el proceso de explotación de datos. En este sentido, un enfoque centrado en el usuario proporciona una mejora en la eficacia, satisfacción de usuario y accesibilidad.

3. Minería de datos centrada en el usuario

En este trabajo se proyecta una minería de datos centrada en el usuario con el propósito de mejorar la experiencia de usuario y tener proyectos funcionales y usables, teniendo como característica la creación de herramientas interactivas centradas en el usuario como apoyo para el proceso de la toma de decisiones. Para esto se incluyeron fundamentos del diseño centrado en el usuario a través de la norma ISO 9241-210:2010, y el proceso CRISP-DM. Se eligió CRISP-DM por ser uno de los principales procesos más utilizados por la comunidad internacional. Estudios recientes [13] destararon que CRISP-DM tiene una mayor aceptación con 43%, comparado con otros procesos, como SEMMA (8.5%) y KDD (7.5%). Se demostró también una alta aceptación de procesos propios, esto es, creados a la medida, con 27.5% de aceptación.

Por otro lado, se eligió la norma ISO 9241-210:2010, la cual es un estándar definido por la Organización Internacional de Normalización (ISO, por sus siglas en inglés), debido a las características, requisitos y recomendaciones que proporciona para el diseño centrado en usuario. Precisamente, estas características sirvieron como referencia para garantizar el diseño centrado en el usuario, a través de cinco etapas

iterativas [17]: a) análisis del contexto de uso, b) especificación de requerimientos, c) producción de soluciones de diseño, d) evaluación del diseño, y e) solución de diseño. En la Fig. 1 se presenta la estructura general de la minería de datos centrada en el usuario con base a los principios de la norma ISO 9241-210:2010 y CRISP-DM.

El objetivo del proceso es involucrar al usuario en etapas significativas de la minería de datos, siguiendo para esto un ciclo iterativo para conocer objetivos, necesidades, actividades, entornos de trabajo, entre otros aspectos, dividido en tres etapas principales (análisis, minería de datos y despliegue) y sus respectivas subetapas.

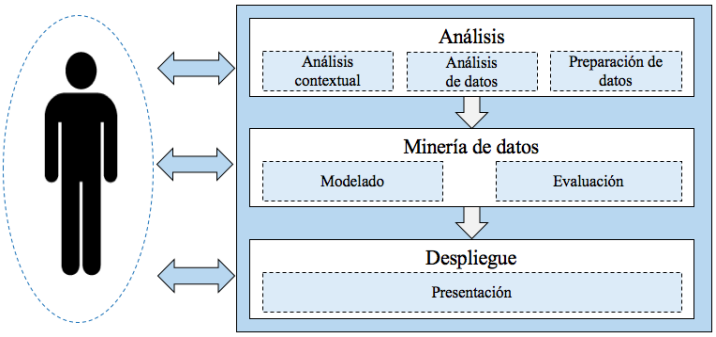

Fig. 1. Estructura general del proceso de minería de datos centrado en el usuario.

En la Fig. 2 se presentan las etapas y actividades que comprende el proceso de minería de datos centrado en el usuario, resaltando el diseño centrado en el usuario.

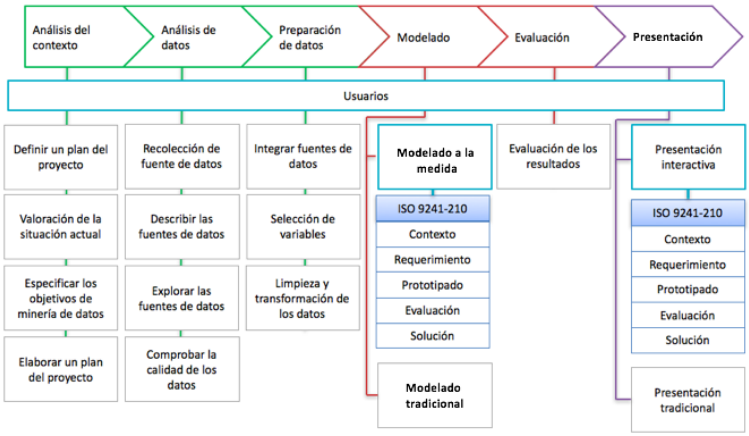

Fig. 2. Tareas generales comprendidas en el proceso de minería de datos centrado en el usuario.

De manera general, las fases que comprenden la minería de datos centrada en el usuario son: a) *análisis contextual*, que engloba en el entendimiento y descripción de los stakeholders, además, se definen los objetivos del proyecto y de minería de datos que se pretenden alcanzar y se elabora un plan general de proyecto; b) *análisis de datos*,

consiste en tener una aproximación sobre el entendimiento de los datos; c) *preparación de datos*, empleada para obtener la vista de datos minable sobre la cual se aplican las técnicas de minería de datos; d) *modelado*, contempla la selección de una o más técnicas de minería de datos para encontrar patrones de datos útiles, tendencias o nuevo conocimiento, en función de las necesidades del proyecto y del usuario (uso de herramientas libres o comerciales, o nuevas aplicaciones personalizadas); e) *evaluación*, consiste en evaluar los resultados desde el punto de vista de los objetivos del proyecto y de los usuarios; y f) *presentación*, comprende la presentación de los resultados obtenidos a través de interfaces interactivas.

4. Cáncer de mama en mujeres de origen mexicano

El cáncer de mama es un tumor maligno que se origina en las células de la mama. Estas células crecen de manera desordenada, logrando invadir tejidos que lo rodean, así como órganos distantes [18], representando en la actualidad una de las tres causas principales de muerte femenina en América Latina [19, 20]. La posibilidad de curación y de mejora en la calidad de vida de las pacientes con cáncer de mama depende de la extensión de la enfermedad en el momento del diagnóstico y de la aplicación adecuada de todos los conocimientos y recursos validados, incrementando la eficiencia y calidad técnica, utilizando para esto la evidencia científica [21]. En este sentido, surge la necesidad natural de propiciar investigaciones desde el punto de vista tecnológico y científico para desarrollar nuevas herramientas de apoyo que sirvan para identificar comportamientos y tendencias de la enfermedad.

Para esta investigación se utilizaron casos de cáncer de mama diagnosticados en mujeres de origen mexicano. La fuente de datos proviene del Programa de Vigilancia, Epidemiología y Resultados Finales (SEER, por sus siglas en inglés) del Instituto Nacional del Cáncer (NCI, por sus siglas en inglés). SEER se encarga de la recopilación de la información sobre los casos de cáncer diagnosticados, sobre las muertes atribuidas a esta enfermedad y la supervivencia de pacientes con cáncer.

En la actualidad, son diversas las investigaciones que se realizan a través del uso de los registros del cáncer, los cuales están a disposición de investigadores, médicos, funcionarios de salud pública, legisladores, políticos, grupos de investigación y público en general [20].

El análisis preliminar de datos identificó un total de 740506 registros y 146 variables, comprendidos entre 1973 y 2012, además, para esta investigación se utilizaron como referencia otros análisis realizados a la base de datos SEER. Estos análisis fueron efectuados bajo modelos matemáticos-estadísticos (análisis correlacional de datos y análisis de componentes principales) y la opinión de especialistas en el campo de la Salud, en los cuales se identificaron 34 variables significativas que tienen relación directa con el cáncer de mama y con registros suficientes en periodos consecutivos [22, 23].

A partir de este análisis se hizo una selección vertical (variables) y horizontal (registros) de los datos, se tomaron en cuenta únicamente variables asociadas al cáncer de mama en mujeres de origen mexicano. Así, la vista de datos minable final quedó

conformada por 16 variables y 2652 registros (Tabla 2), tomando como variable clase el estado de vida del paciente (Vital Status recode) cuyos valores binarios son: 0 para la mortalidad y 1 para la supervivencia.

Tabla 2. Variables seleccionadas para la vista de datos minable.

#	Nombre de variable	Descripción	Tipo
1	Marital Status at DX	Estado civil	Discreto
2	Age at diagnosis	Edad del paciente	Discreto
3	Month of diagnosis	Mes de diagnostico	Discreto
4	Year of diagnosis	Año de diagnostico	Discreto
5	Laterality	Lado donde se originó el tumor	Discreto
6	Behavior Code ICD-O-3	Tipo de comportamiento de la neoplasia	Discreto
7	Grade	Clasificación de las células cancerígenas	Discreto
8	Diagnostic Confirmation	Método de confirmación del cáncer	Discreto
9	Regional Nodes Examined	Numero ganglios linfáticos removidos y examinados	Discreto
10	RX Summ-Radiation	Método de radioterapia llevado a cabo	Discreto
11	RX Summ-Surg / Rad Seq	Secuencia de la cirugía y radioterapia	Discreto
12	Age Recode <1 Year olds	Grupo de edad(intervalos de 5 años)	Discreto
13	Survival Months	Tiempo de supervivencia del paciente(meses)	Discreto
14	Tumor Size	Tamaño del tumor	Discreto
15	AJCC Stage	Etapa de la enfermedad	Discreto
16	Vital Status recode	Estado de vida del paciente	Binario

Para este trabajo se desarrolló una aplicación (Fig. 3) con base en el diseño centrado en el usuario, la cual quedó integrada por cuatro secciones principales: a) Operadores, contiene funciones para cargar la fuente de datos, seleccionar la vista de datos minable, seleccionar el algoritmo de minería de datos y validar su precisión; b) Diseño de minería de datos, permite esquematizar una secuencia de operadores para la ejecución de los algoritmos de minería de datos c) Configuración de operadores, permite configurar los operadores en la sección de Diseño; y d) Resultados, presenta los resultados obtenidos a través de una interfaz interactiva.

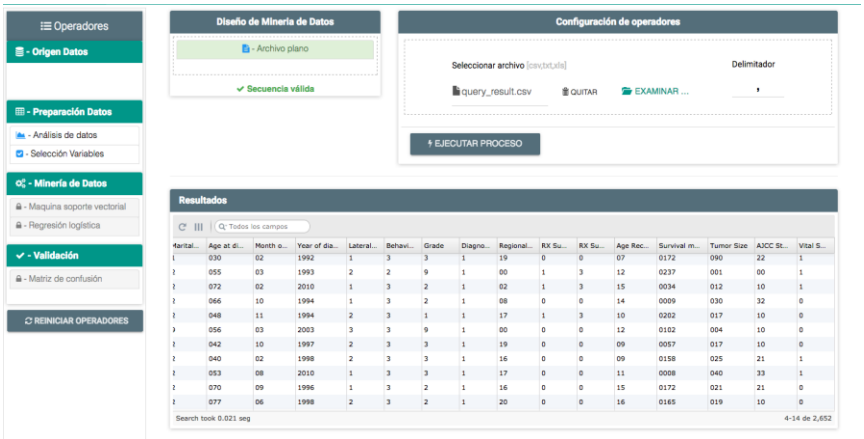

Fig. 3. Interfaz de la aplicación de minería de datos centrada en el usuario.

Por otra parte, con la finalidad de que el usuario cometa la menor cantidad de errores en el uso de la aplicación, se diseñó e implementó un autómata de usabilidad para

validar la secuencia correcta en la colocación de los operadores en la sección de Diseño. Para esto se definieron funciones para habilitar y deshabilitar los operadores con el propósito de mejorar la usabilidad y la experiencia del usuario. Los estados del autómata representan nodos secuenciales (Fig. 4).

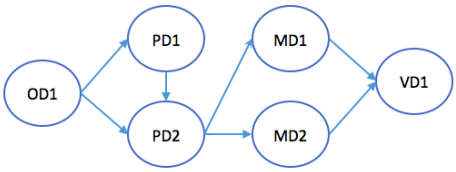

Fig. 4. Autómata para validar la ejecución de la secuencia de los operadores.

- Archivos planos (OD1). Tiene la funcionalidad de cargar y mostrar el conjunto de datos de tipo plano, con formatos csv, .txt o .xls.
- Análisis de datos (PD1). Permite analizar el conjunto de datos a través de tipos de gráficas diversas, por ejemplo, líneas, puntos, barras y otros.
- Selección de variables (PD2). Permite hacer una configuración de la entrada y salida de las variables que forman parte la vista de datos minable.
- Máquina de soporte vectorial (MD1). Con este operador se hacen las predicciones de una o más variables clase, tomando como entrada la vista de datos minable.
- Regresión logística (MD2). Este operador permite la predicción de una o más variables clase con base en las variables predictoras.
- Matriz de confusión (VD1). Mediante esta función se hace la evaluación de la precisión de los algoritmos de clasificación.

La validación de la secuencia de los operadores (Fig. 5 y 6) permitió guiar al usuario en el diseño de obtención de patrones, previniendo acciones fallidas. Por ejemplo, para el caso de la secuencia no válida, ésta se produce debido a que una vez cargado el conjunto de datos (OD1) es necesario definir las variables independientes, así como la variable clase (PD2), esto siguiendo el tipo de aprendizaje supervisado, y no utilizar antes el algoritmo de minería de datos (MD1 o MD2).

Cabe mencionar que una vez ejecutado algún algoritmo de minería de datos, es posible hacer una reconfiguración de las variables y conjunto de datos para obtener nuevos resultados. Es importante destacar también que a medida que se necesite incluir nuevos operadores a la herramienta, el autómata permite anexar nuevos nodos haciendo que el software sea escalable.

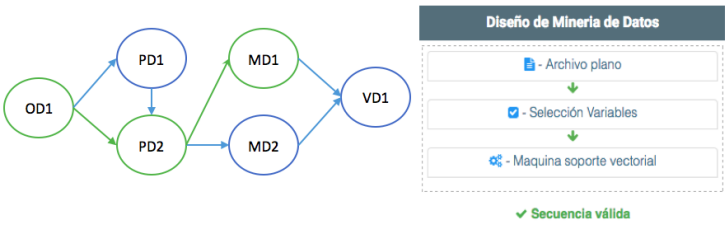

Fig. 5. Secuencia válida detectada por el autómata.

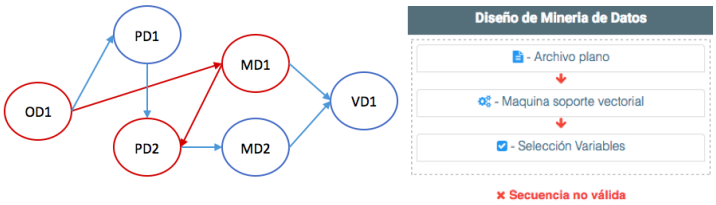

Fig. 6. Secuencia no válida detectada por el autómata.

Por otra parte, para ofrecer un mejor uso de la aplicación se realizaron pruebas de usabilidad con el propósito de hacer mejoras en la interfaz de usuario. Se trabajó con ocho usuarios con conocimientos en minería de datos. Se utilizó como método las tareas guiadas, esto es, se dictó a los usuarios en voz alta las tareas que debían realizar.

Los resultados alcanzados se muestran en la Fig. 7. Se identificó que seis usuarios concluyeron correctamente todas las tareas; otros dos tuvieron un error al completar una de las tareas asignadas. Se detectó además que en la etapa de selección de variables no había suficiente información para realizar la tarea. Esto permitió hacer las correcciones en la aplicación. Una vez construida la aplicación y evaluada desde el punto de vista de la usabilidad se hizo la ejecución para analizar los casos diagnosticados de cáncer de mama en pacientes mujeres de origen mexicano.

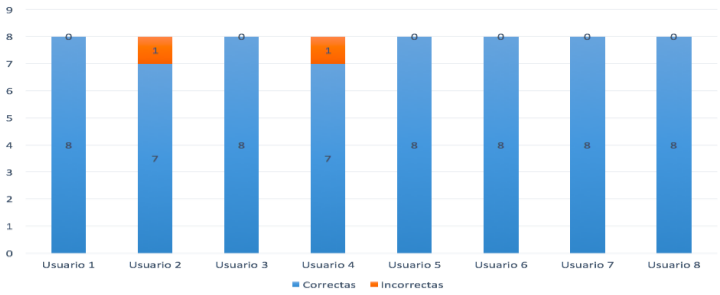

Fig. 7. Tareas correctas e incorrectas realizadas por los usuarios.

Posterior a la ejecución de los algoritmos, máquina de soporte vectorial y regresión logística, se evaluaron los resultados a través de una matriz de confusión mediante la cual se obtuvieron precisiones de clasificación de 87.4 y 85.6 %, respectivamente (Tabla 3). Se observó además que los resultados obtenidos para la supervivencia y mortalidad (Fig. 8 y 9) siguen un patrón similar al de los datos originales.

Tabla 3. Resultados de la precisión obtenidas por los algoritmos de clasificación.

Algoritmo	Total de casos	Correctos Positivos	Correctos Negativos	Falsos Positivos	Falsos Negativos	Precisión
Regresión logística	2,652	1695	624	239	94	87.4 %
Máquina de soporte vectorial	2,652	1725	547	316	64	85.6 %

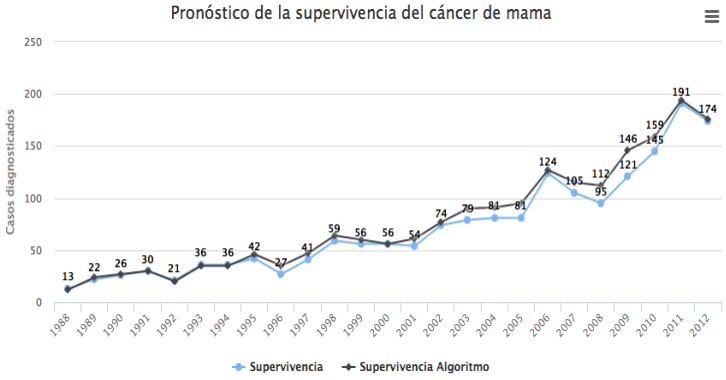

Fig. 8. Clasificación de la supervivencia del cáncer de mama.

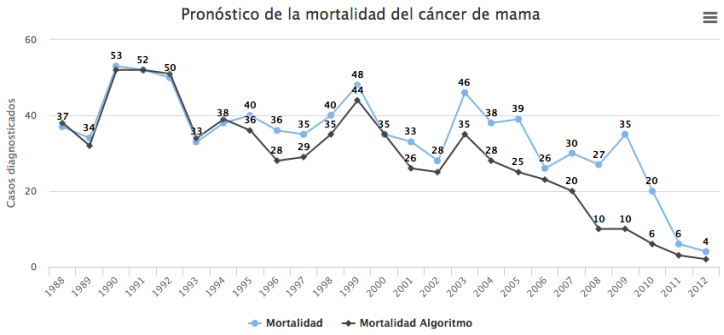

Fig. 9. Clasificación de la mortalidad del cáncer de mama.

Se calcularon otros parámetros de validez, como: sensibilidad y especificidad. La sensibilidad es la probabilidad de clasificar correctamente a un individuo vivo, es decir, la probabilidad de que una persona viva sea clasificada como un verdadero positivo (supervivencia), en este caso se obtuvo un valor de 95%, y la especificidad que es la probabilidad de clasificar correctamente a una persona muerta, es decir, una persona muerta sea clasificada como verdadero negativo (mortalidad), con 72%.

Para la presentación de los patrones de datos obtenidos se implementó una interfaz interactiva (Fig. 10), facilitando al usuario un mejor entendimiento de los resultados obtenidos a través de una serie de gráficas e interacciones definidas con los usuarios finales (médicos). Estas gráficas interactivas están relacionadas con las variables oncológicas de interés para los médicos especialistas del Hospital General La Raza (Cd. México). Estas variables son (descritas en la Tabla 2): Laterality, Behavior Code ICD-O-3, Grade, Diagnostic Confirmation y RX Summ-Radiation.

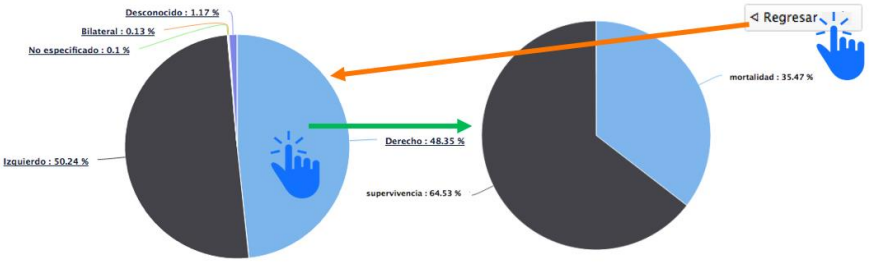

Fig. 10. Muestra de una gráfica interactiva de la lateralidad del tumor.

Una parte importante en el proyecto fue la evaluación de la presentación de los resultados a los médicos especialistas. Para esta evaluación fueron tomados en cuenta cuatro especialistas, quienes proporcionaron los requisitos para el desarrollo de la aplicación. La primera evaluación fue a través del método SIRIUS [24] obteniendo un porcentaje de usabilidad de 93.88%, 87.86%, 96.22%, 90.94%, respectivamente; y como segunda parte de evaluación se les presentó un checklist de verificación de usabilidad, sugerida en la Guía para Desarrollo de Sitios Web [25] con base a la heurísticas de Nielsen [26], obteniendo resultados con respecto a las siguientes heurísticas: a) navegación, todos los usuarios estuvieron de acuerdo con sus respuestas, logrando así una satisfacción positiva perfecta; b) visibilidad del estado del sistema, estética y diseño, retroalimentación, obtuvieron una satisfacción positiva alta de 3.25, 3.75 y 3 respectivamente; y c) ayuda ante errores, obtuvo una satisfacción baja con un valor de 1.25.

5. Conclusiones

La integración de un método centrado en el usuario en un proceso clásico de minería de datos no sólo fue para involucrar al usuario, sino también aspectos del diseño centrado en el usuario para el desarrollo de aplicaciones personalizadas, esto es, soluciones implementadas a la medida (Ad hoc). Esto se logró verificar mediante la experimentación práctica sobre la clasificación de la supervivencia y mortalidad de mujeres de origen mexicano diagnosticadas con cáncer de mama.

Como fase inicial del estudio se hizo la adquisición de la fuente de datos, correspondientes a registros clínicos de casos de cáncer de mama en mujeres de origen mexicano. Esta fuente de datos fue adquirida a través de un acuerdo de confidencialidad. Así, dada las características propias de la fuente de datos fue necesario hacer un tratamiento de ésta con el propósito de hacer una selección de datos significativos, vertical y horizontal. Producto de esto se generó una vista de datos conformada por 16 variables y 2652 registros.

Producto de la ejecución de los algoritmos implementados se obtuvo precisiones de 85.6% (máquina de soporte vectorial) y 87.4% (regresión logística). Esto indica que la clasificación de los casos de supervivencia y mortalidad asociados al cáncer de mama pueden ser pronosticados con una precisión notable, con una sensibilidad del 95% y

especificidad 72%, evidenciando la utilidad de los resultados obtenidos para el proceso de la toma de decisiones en el contexto médico.

Las pruebas de usabilidad realizadas sobre el prototipo, basadas en los métodos SIRIUS y Checklist, permitieron identificar mejoras sobre la interacción entre la aplicación y los usuarios.

Agradecimiento. Este trabajo forma parte del proyecto "Infraestructura para agilizar el desarrollo de sistemas centrados en el usuario" financiado por el Consejo Nacional de Ciencia y Tecnología, en el marco de Cátedras CONACYT (Ref. 3053).

Referencias

1. Zhao, Y., Chen, Y., Yao, Y.: User-centered interactive data mining. In: Cognitive Informatics, 5th IEEE International Conference, 457–466 (2006)
2. Horberry, T., Burgess-Limerick, R., Steiner, L.: Human Centred Design for Mining Equipment and New Technology. In: Proceedings 19th Triennial Congress of the IEA, 9, 14 (2015)
3. Brachman, R., Anand, T.: The process of knowledge discovery in databases. In Advances in knowledge discovery and data mining, American Association for Artificial Intelligence (1996)
4. Haun, S., Nürnberger, A.: Supporting exploratory search by user-centered interactive data mining. In: SIGIR Workshop Information Retrieval for E-Discovery (SIRE) (2011)
5. Habib ur Rehman, M., Liew, C. S., Wah, T. Y.: UniMiner: Towards a unified framework for data mining, Information and Communication Technologies (WICT), 134–139 (2014)
6. ISO 9241-210:2010.: Ergonomics of human system interaction-Part 210: Human-centred design for interactive systems.International Standardization Organization (ISO) (2010)
7. Nigro, H. O., Gonzáles, S.E., Xodo, D. H.: Data Mining with Ontologies: Implementations. Findings, and Frameworks (2007)
8. Moine, J., Gordillo, S., Haedo, A.: Análisis comparativo de metodologías para la gestión de proyectos de Minería de Datos. Workshop Bases de Datos y Minería de Datos, 931–938 (2011)
9. SAS Institute.: Data Mining and the Case for Sampling. Data Mining Using SAS Enterprise Miner (2015), http://sceweb.uhcl.edu/boetticher/ML_DataMining/SAS- SEMMA.pdf
10. Sumathi, S., Sivanandam, S.: Introduction to Data Mining and its Applications. Studies in Computational Intelligence, 29, editado por Springer-Verlag, Heidelberg, Alemania (2006)
11. Peralta, F.: Elementos para un mapa de actividades para proyectos de explotación de información. Facultad Regional Buenos Aires, Argentina 52 (2013)
12. Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., Wirth, R.: CRISP-DM 1.0 Step-by-step Data Mining Guide,74 (2000)
13. KDnuggets: Data Mning, Analytics, Big Data, and Data Science. http://kdnuggets.com/2014/10/crisp-dm-top-methodology-analytics-data-mining-data-science-projects.html (2014)
14. Rivo, E., de la Fuente, J., Rivo, Á., García-Fontán, E., Cañizares, M., Gil, P.: Cross-Industry Standard Process for data mining is applicable to the lung cancer surgery domain. Clinical and Translational Oncology, 14(1), 73–79 (2012)
15. Pyle D.: Business modeling and data mining. Morgan Kaufmann 720 (2003)

16. Britos P. V.: Procesos de explotación de información basados en sistemas inteligentes. Universidad Nacional de la Plata, Buenos Aires, Argentina 234 (2008)
17. Jang, G., Jeon, J.: A Six Sigma Methodology Using Data Mining: A Case Study on Six Sigma Project for Heat Efficiency Improvement of a Hot Stove System in a Korean Steel Manufacturing Company. Research Topics on Multiple Criteria Decision Making, 72-80 (2009)
18. IMSS: Cáncer de mama. http://imss.gob.mx/salud-en-linea/cancer-mama (2016)
19. INEGI: Estadísticas a propósito del día mundial de la lucha contra el cáncer de mama. http://inegi.org.mx/saladeprensa/aproposito/2015/mama0.pdf
20. NCI: Cáncer de mama: Información general sobre el cáncer de mama. http://cancer.gov/espanol/tipos/seno (2016)
21. Secretaria de Salud: Diagnóstico y Tratamiento del Cáncer de Mama en Segundo y Tercer nivel de Atención (2009)
22. Molero G., Céspedes Y., Meda M.: Caracterización y análisis de la base de datos de cáncer de mama SEER-DB. IX Congreso Internacional Informática en Salud (2013)
23. Molero G.: Clasificador bayesiano para el pronóstico de la supervivencia y mortalidad de casos de cáncer de mama en mujeres de origen hispano (Tesis doctoral). Universidad de Guadalajara, México 155 (2014)
24. Suárez, M.: SIRIUS: Sistema de Evaluación de la Usabilidad Web Orientado al Usuario y basado en la Determinación de Tareas Críticas (2011)
25. Guiadigital.: Checklist de Usabilidad.http://guiadigital.gob.cl/articulo/usabilidad-0, (2016)
26. Nielsen, J.: Usability engineering. Academic Press Limited, Massachuetts, Estados Unidos, 361 (1993)

Impreso en los Talleres Gráficos
de la Dirección de Publicaciones
del Instituto Politécnico Nacional
Tresguerras 27, Centro Histórico, México, D.F.
noviembre de 2016
Printing 500 / Edición 500 ejemplares