Advances on Language & Knowledge Engineering

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 144,** noviembre 2017. 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. Queda prohibida la reproducción total o parcial, por cualquier medio, sin el permiso expreso del editor, excepto para uso personal o de estudio haciendo cita explícita en la primera página de cada documento. Impreso en la Ciudad de México, en los Talleres Gráficos del IPN – Dirección de Publicaciones, Tres Guerras 27, Centro Histórico, México, D.F. Distribuida por el Centro de Investigación en Computación, Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othón de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738, México, D.F. Tel. 57 29 60 00, ext. 56571.

Editor responsable: *Grigori Sidorov, RFC SIGR651028L69*

Research in Computing Science is published by the Center for Computing Research of IPN. **Volume 144**, November 2017. Printing 500. The authors are responsible for the contents of their articles. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without prior permission of Centre for Computing Research. Printed in Mexico City, in the IPN Graphic Workshop – Publication Office.

Volume 144

Advances on Language & Knowledge Engineering

Beatriz Beltrán
Darnes Vilariño
Josefa Somodevilla
David Pinto

SEP
SECRETARÍA DE EDUCACIÓN PÚBLICA

Instituto Politécnico Nacional
“La Técnica al Servicio de la Patria”

Instituto Politécnico Nacional, Centro de Investigación en Computación
México 2017

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

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

Indexed in LATINDEX, DBLP and Periodica

Printing: 500

Printed in Mexico

ISSN 1870-4069

Editorial

This volume of the "Research in Computing Science" journal contains selected papers on recent Advances on Language and Knowledge Engineering. The papers were carefully chosen by the editorial board on the basis of the at least two double blind reviews by the members of the reviewing committee or additional reviewers. The reviewers took into account the originality, scientific contribution to the field, soundness and technical quality of the papers. It is worth noting that various papers for this special issue were rejected (rejected rate was 25%).

The volume contains regular papers and a selection of papers presented in the 5th International Symposium on Language & Knowledge Engineering (LKE'2017), an academic conference organized in the Faculty of Computer Science of the Benemérita Universidad Autónoma de Puebla (BUAP) which has been created and organized for the fourth time by the Language & Knowledge Engineering Lab with the aim of offering an academic platform in which experts in related areas may exchange experiences and publish their recent research advances.

We would like to thank Mexican Society for Artificial Intelligence (Sociedad Mexicana de Inteligencia Artificial) and the Thematic Academic Network named "Language Technologies" (Red Temática en Tecnologías del Lenguaje) for their invaluable support in the construction of this volume.

The entire submission, reviewing, and selection process, as well as preparation of the proceedings, were supported for free by the EasyChair system (www.easychair.org).

Beatriz Beltrán
Darnes Vilariño
Josefa Somodevilla
David Pinto

Guest Editors

Benemérita Universidad Autónoma de Puebla,
LKE-FCC-BUAP, Mexico

November 2017

ISSN 1870-4069

Table of Contents

Page

ISSN 1870-4069

Extracción automática de eventos indicadores a partir de noticias en español

Ariatna Quinto[1], Belém Priego[1], David Pinto[2], José A. Reyes-Ortiz[1]

[1]Universidad Autónoma Metropolitana unidad Azcapotzalco,
Departamento de Sistemas, México

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

{al2113034676,abps,jaro}@azc.uam.mx,dpinto@cs.buap.mx

Resumen. Actualmente, se ha notado el incremento exorbitante de la información electrónica, claro ejemplo es la Web, la cual se ha convertido de fácil acceso. Ésta es posible estudiarla para saber los fenómenos que suceden a nivel de la lengua y a partir de ella se puede extraer meta información. En este artículo se han seleccionado artículos periodísticos en formato electrónico, como corpus, para llevar a cabo la extracción automática de eventos indicadores que representan la proporcionalidad de un evento con respecto a un total, es decir, porcentajes relacionados a algún suceso. El principal objetivo de esta investigación es mostrar estadísticas que ocurren alrededor del mundo mediante la búsqueda de patrones lingüísticos en un sistema de recuperación de información. Este artículo presenta los avances obtenidos hasta el momento, en la extracción automática de eventos indicadores a partir de noticias en español.

Palabras clave: Extracción automática, recuperación de información, eventos indicadores.

Automatic Extraction of Indicative Events from Spanish News Stories

Abstract. Nowadays there exist an increasing raising of information on Internet which is easy available for human beings. This huge volume of data can be analyzed in order to discover and model linguistic phenomena and extract information. In this paper we have selected the news stories genre for extracting indicative events that represent likelihood of some event to occur. The aim of this research is to bring to light statistical events occurring all around the world by employing techniques of natural language processing based on linguistic patterns in an information retrieval system. This paper presents the outcomes obtained up to now in the particular topic of automatic extraction of indicative events from Spanish news stories.

Keywords. Automatic extraction, information retrieval, indicative events.

pp. 9–17; rec. 2017-07-09; acc. 2017-10-20

1. Introducción

Procesamiento de Lenguaje Natural (denotado por PLN), es una disciplina de la Inteligencia Artificial que trata la formulación e investigación de mecanismos de computación para la comunicación entre personas y máquinas, mediante el uso de Lenguajes Naturales. Dichos lenguajes son utilizados para la comunicación ya sea de forma escrita, hablada o en forma de signos [4]. Entre las tareas que realiza el PLN se encuentra la extracción automática de eventos, cuyo objetivo es capturar ciertas partes relevantes de un texto.

En el análisis del lenguaje se estudia la estructura del lenguaje a cuatro niveles [4]:

- Análisis morfológico: El análisis de las palabras para extraer raíces, rasgos flexivos, unidades léxicas compuestas y otros fenómenos.
- Análisis sintáctico. El análisis de la estructura sintáctica de la frase mediante una gramática de la lengua en cuestión.
- Análisis semántico. La extracción del significado (o posibles significados) de la frase.
- Análisis pragmático. El análisis de los significados más allá de los límites de la frase, por ejemplo, para determinar los antecedentes referenciales de los pronombres.

Este artículo, pretende abordar la tarea de la extracción automática de eventos. En particular, los eventos serán los indicadores que representan la proporcionalidad de un evento con respecto a un total que aparecen en una colección de información (noticias) en español, es decir, éstos son los porcentajes relacionados con algún suceso. Para el desarrollo total de esta investigación se hará uso únicamente de tres niveles del lenguaje, el morfológico (etiquetado POS), el sintáctico (segmentación del párrafo en oraciones) y el semántico (patrones lingüísticos). Además, se ha seleccionado el género periodístico debido a que es un tipo de escritura estándar y homogénea, en otras palabras, cualquier hablante nativo que lea el contenido de una nota periodística, lo entiende. Inclusive, la mayoría de las personas tiene acceso a un periódico, en formato papel o digital. Sin embargo, dado que es una investigación que está iniciando en este artículo, se presentarán los avances obtenidos al tratar de extraer de manera automática eventos indicadores a partir de noticias en español.

2. Motivación

Debido a la basta cantidad de información que actualmente se encuentra en la Web, ésta se puede utilizar y procesar de modo que se pueda emplear para ciertas tareas del PLN, una de ellas es la extracción de información relevante de un texto. Además de que es posible extraer conocimiento de toda esta información. Uno de los medios de comunicación que proporciona información es el Periódico, que especialmente en los últimos años su acceso ha sido en formato digital; esto debido a los avances tecnológicos, como es internet. Razón por la cual, en esta investigación se ha decidido trabajar con este género textual.

A través de los periódicos se ha podido llegar incluso a más gente y mantener un ritmo de actualización de los datos mucho más intenso que antes, siendo hoy imposible esperar de un día para otro para conocer noticias. Lo interesante de los periódicos es que cuando se habla de una sociedad, más o menos compleja, se pueden encontrar distintos tipos de periódicos que dan con el perfil de grupos sociales particulares, de grupos de edad, de regiones geográficas, de actividades laborales, de intereses específicos como deportes, internacionales, espectáculos o política.

Debido a que la sociedad cada vez vive de una manera más acelerada, se ha decidido extraer información de las noticias y a través de esta extracción dar a conocer datos estadísticos que la búsqueda de eventos indicadores proporcionará como resultado. Lo que servirá para informar de manera más concisa y directa cierto suceso, esta información representará la proporcionalidad de un evento, sin la necesidad de que las personas lean toda la nota periodística, o más notas, para conocer datos importantes y relevantes. Al finalizar la investigación, se pretende crear un sistema web capaz de mostrar los resultados de la extracción de eventos indicadores. Dicho sistema servirá como medio de información resumida, lo cual beneficiará a la sociedad que desea estar informada.

3. Descripción del problema

En el Diccionario de la Real Academia [9], una de las acepciones para “evento” dice que es un suceso importante. De esta manera, este artículo retoma esa idea para mostrar un suceso importante que surge alrededor del mundo a través de los relatos periodísticos que se encuentran en la web. Sin embargo, se sabe que dar una definición concisa de evento es difícil. Por lo tanto, en esta investigación se tratará a un suceso como cualquier tipo de situación o acontecimiento que ocurre, restringiendo a eventos relacionados con la proporcionalidad de uno con respecto a un total, es decir porcentajes relacionados a un suceso. En (1) y (2) se pueden observar enunciados que muestran eventos indicadores.

(1) En México, cerca del 88 % de la energía primaria que se consume proviene del petróleo.

(2) La mitad de la población mundial está concentrada en tan solo seis países.

En (1) el evento indicador de porcentaje está explícitamente determinado por su signo ortográfico. Mientras que en (2) se nota un evento indicador, la mitad de la población mundial, pero no está explícitamente denotado por un signo ortográfico. Por lo que la tarea de extracción de eventos indicadores se convierte más complicada a medida que el lenguaje se desarrolla con diferentes factores de pensamiento, es decir, a medida que el lenguaje es abundante, enriquecido y creativo, la tarea se complica.

Típicamente, los eventos pueden ser expresados por verbos conjugados o en infinitivo, predicados en general y frases preposicionales. Sin embargo, como se ha notado, en esta investigación no se cumple con esos acuerdos, ya que se pueden encontrar eventos expresados mediante un signo de puntuación (%), partes que dividen un todo (mitad, tercio, etc.), entre otras expresiones. Sobre un texto plano, se pretende identificar eventos indicadores mediante diferentes patrones lingüísticos que serán identificados y son los que ayudaran a la extracción de diferentes sucesos acontecidos alrededor del mundo. Se puede delimitar entre las etiquetas <*indicador*> y </*indicador*>, el inicio y fin del indicador porcentual con el fin de poder extraer toda la idea que contiene al evento, en (3) se puede observar un ejemplo. Esto mediante un sistema que segmente a los párrafos en oraciones.

(3) La <*indicador*>mitad</*indicador*> de la población mundial está concentrada en tan solo seis países.

En el contexto de este trabajo se contará con un esquema que establece reglas claras con guías de cómo se deben identificar los eventos indicadores a partir de diferentes patrones lingüísticos, con el objetivo de reducir las ambigëdades al mínimo.

4. Estado del arte

En esta sección, se describen los trabajos reportados en la literatura relacionados con la extracción automática de eventos. Cabe mencionar, que se incluyen trabajos que extraen eventos, sin embargo, corresponden a otro tipo que difiere al presentado en este trabajo, pero se incluyen debido a la temática presentada.

La utilización de características sintácticas, semánticas y contextuales, minería de textos, es uno de los varios enfoques que se utilizan para la extracción de eventos. El trabajo de Hernández [3], aborda su investigación sobre dicho enfoque, diseñó e implementó un sistema web para la anotación semántica de actores y eventos a partir de un corpus de textos periodísticos mexicanos. Este trabajo tiene la particularidad principal, que la extracción la realiza en textos periodísticos mexicanos, sin embargo, son relatos periodísticos completamente diferentes a los que se abordan en este trabajo. El trabajo de Gil-Vallejo et al. [2], incluido en este enfoque, extrae eventos mediante las entidades y relaciones que posee el texto usando información lingística, relaciones sintácticas y semánticas, obteniendo un patrón de extracción de información relevante para un cierto evento. Los eventos que considera son los que contienen verbos dentro de su estructura de frase.

Las técnicas de aprendizaje automático, también, son consideradas para el reconocimiento de eventos. Un trabajo presentado por Moncecchi y Rosá [5], que entra en el marco de este enfoque, reconoce eventos en textos españoles utilizando como base el esquema de anotación SIBILA [14] y dos algoritmos de aprendizaje automático, campo aleatorio condicional (CRF, por sus siglas en inglés Conditional Random Fields) [11] y máquinas de soporte vectorial (SVM, por sus siglas en inglés Support Vector Machines) [10], logrando mejores resultados con SVM.

Además de investigaciones, existe el desarrollo de herramientas capaces de extraer eventos en un texto. JASPER [1] y TES [12], son un claro ejemplo de ello.

En la primera, JASPER: Journalist's Assistant for Preparing Earnings Reports [1], se extraen ciertas piezas clave de información de un rango limitado de texto. El sistema está basado en el uso de plantillas, técnicas de comprensión parcial y procedimientos heurísticos para extracción de información. Esta información, puede ser utilizada de varias maneras, como rellenar valores en una base de datos, generar resúmenes del texto de entrada, entre otras. Los eventos que principalmente extrae JASPER son los comunicados de prensa para generar historias de noticias. Con respecto a la segunda, TES: Terminology Extraction Suite [13], es una herramienta desarrollada para la extracción automática de terminología, que permite obtener términos y buscar automáticamente equivalentes de traducción. La herramienta está escrita en Perl, con interfaces gráficas implementadas en Tk. En este caso, los términos son eventos o sucesos que un documento, texto, contiene.

5. Metodología propuesta

En la extracción automática de eventos indicadores a partir de noticias en español, se han identificado dos etapas principales (ver Figura 1), las cuales serán alimentadas por un corpus de notas periodísticas en texto plano escritas en español,. A partir de estas notas se desarrollará la extracción automática de los eventos indicadores. La extracción se llevará a cabo mediante el diseño de patrones lingüísticos que cubrirán tres niveles de la lengua, morfológico, sintáctico y semántico. Finalmente, se pretende visualizar los resultados en un sistema web.

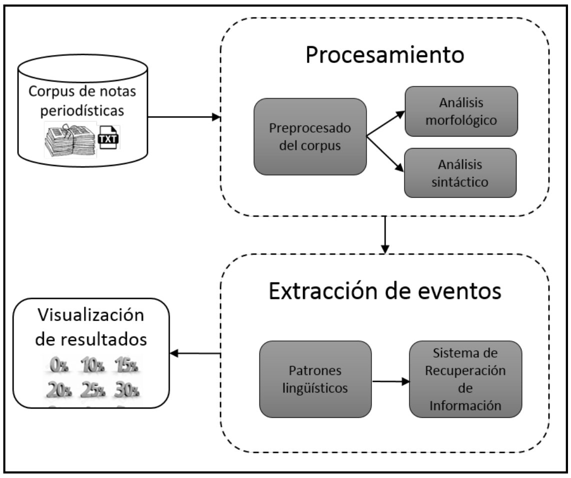


Fig. 1. Metodología propuesta para la extracción automática de eventos indicadores.

5.1. Conjunto de datos

En esta sección se describe el conjunto de datos, corpus periodístico, en español de notas periodísticas que será utilizado para la extracción automática de eventos indicadores. Cabe mencionar que la descripción realizada es general debido a que el corpus a utilizar es el realizado en [9].

El corpus ha sido extraído del sitio de intenet de la Organización Editorial Mexicana[1], OEM, que contiene relatos periodísticos escritos en español mexicano. A pesar de ser un sitio mexicano, no excluye las notas periodísticas internacionales pero de igual manera escritas en español mexicano. Los relatos periodísticos corresponden al período de tiempo del año 2007 al 2013.

Si bien, el corpus presenta diferentes metadatos, para el caso de este artículo, sólo se considerará el texto plano de la nota periodística. El corpus utilizado para esta tarea consta de 378,890 noticias, un total de 4,579,284 oraciones y alrededor de 11,1595,71 palabras.

5.2. Etapa de procesamiento

La etapa de procesamiento comprende el preprocesado del corpus, el análisis morfológico y sintáctico de dicho conjunto de datos.

La primera actividad, el preprocesado del corpus, contempla la eliminación de signos de puntuación y de palabras cerradas, mediante la utilización de un lexicón de éstos [8]. Además, se eliminan los caracteres especiales, poniendo atención a los caracteres relacionados con los eventos indicadores (%); estos caracteres a eliminar, se consideran irrelevantes para la extracción de los e entos indicadores debido a que ocasionarán conflicto al momento de recuperar la información que se está extrayendo y podrían incrementar el tiempo de respuesta.

La segunda actividad, el análisis morfológico, realiza el etiquetado de las partes de la oración (PoS de sus siglas en inglés, Part of Speech) en las notas periodísticas; se hace uso de las herramientas FreeLing [6] y/o TreeTagger[13]. El etiquetado consiste en identificar la categoría gramatical de cada palabra y asignarle una etiqueta dependiendo de la categoría gramatical a la que corresponda.

La tercera actividad, el análisis sintáctico, busca segmentar los párrafos que componen al corpus periodístico en oraciones, debido a que será más accesible la manipulación de oraciones y éstas tendrán una longitud más regular, es decir, el tamaño de un párrafo tiene menos proporcionalidad con respecto a una oración.

5.3. Etapa de extracción de eventos

La etapa de extracción de eventos incluye dos actividades esenciales, el diseño de los patrones lingísticos, para la extracción automática de eventos indicadores, y la búsqueda de éstos en un sistema de recuperación de información (denotado por SRI).

[1] Para más información sobre la Organización Editorial Mexicana consultar: https://www.oem.com.mx/oem/

La primera actividad, diseño de patrones lingísticos, permite realizar el diseño de los patrones lingísticos que, en una etapa posterior, se implementan con el fin de extraer los eventos indicadores. Este diseño de patrones permite descubrir los elementos lingüísticos empleados con frecuencia en las notas periodísticas, para ello se utiliza uno o varios modelos que sirven como muestra para identificar y agrupar los eventos indicadores.

La segunda actividad, búsqueda de los patrones lingísticos en un SRI, posibilita, una vez que se han identificado los patrones lingüísticos, la implementación de éstos. Es decir, mediante un SRI, alimentado con el corpus de notas periodísticas y como consulta los patrones lingüísticos identificados, extraer la información relevante relacionada a los eventos indicadores.

6. Resultados obtenidos

El proceso de extracción de patrones lingüísticos se ha llevado a cabo mediante una técnica conocida como bootstrapping. Se considera un conjunto inicial de muestras etiquetadas manualmente, las cuales son posteriormente enriquecidas usando muestras similares obtenidas mediante un sistema de recuperación de información. De esta manera, es posible obtener un conjunto considerable de datos que comparten una estructura morfosintáctica que permite obtener patrones lingísticos que muestran la regularidad de estructuras para un tipo de expresión lingística en particular.

En la Tabla 1 se muestran ejemplos de los patrones lingüísticos más comunes encontrados mediante este proceso de generalización basado en un conjunto inicial moderadamente pequeño de muestras manualmente etiquetadas, pero que fue enriquecido mediante la técnica anteriormente mencionada.

Tabla 1. Patrones lingüísticos más frecuentes para eventos indicadores usando explícitamente el porcentaje.

Patrón lingüístico	Semántica
NUM por ciento	Indicador puntual
alrededor del NUM por ciento	Indicador aproximado
mayor que NUM por ciento	Indicador de punto base
entre NUM y NUM por ciento	Intervalo
de NUM a NUM por ciento	Incremento
hasta un NUM por ciento	Máximo temporal
NUM por ciento más que	Indicador comparativo incremental
un incremento de NUM por ciento	Indicador comparativo incremental
NUM por ciento menos que	Indicador comparativo decremental
un decremento de NUM por ciento	Indicador comparativo decremental

Es importante aclarar que la etiqueta **NUM** se refiere a la especificación de un número en el texto periodístico en cualquiera de sus expresiones. Ejemplos de **NUM** serían los siguientes: 90, 28.3, noventa y tres, etc.

Por otro lado, en la Tabla 1, se ha usado el texto "por ciento", el cual puede ser encontrado también sustituido por el símbolo ortográfico "%" por lo que el lector debe considerar que los patrones anteriormente mencionados pueden ocurrir con cualquiera de estas dos expresiones textuales.

En la Tabla 2, se muestran otros patrones morfosintácticos que expresan el uso de números fraccionarios porcentuales, pero expresados en lenguaje natural.

Tabla 2. Patrones lingüísticos más frecuentes para eventos indicadores que usan otro tipo de expresiones del lenguaje natural relacionadas implícitamente con un porcentaje.

Patrón lingüístico	**Semántica**
un total de NUM de las/los NUM	Indicador porcentual basado en cociente
la mitad	Indicador 50 %
un tercio	Indicador 33 %
una cuarta parte	Indicador 25 %

Ejemplos de oraciones que contienen a algunos de los patrones presentados se muestran en la Tabla 3.

Tabla 3. Ejemplos de eventos indicadores.

Oraciones del género periodístico con un evento indicador
Una cuarta parte de hogares poblanos apenas tiene acceso a Internet
Sólo un tercio -31.8 por ciento- de los poblanos consideró al gobierno municipal
Juntos canjearon 39 planillas, la mitad con nombre de Eugenio, la otra ...
Ventas de comercio al por menor crecerán hasta 7 por ciento en 2016
La economía informal contribuyó con el 24.8 por ciento del Producto Interno Bruto
Esto supone entre un 25 y un 30 por ciento de los ingresos
Nuevo Sistema de Justicia Penal, al 90 % en Michoacán
El 75 % de los trabajadores en México está sometido a algún grado de estrés laboral, y eso a la larga es la causa del 25 % de los 75 mil infartos ...

7. Conclusiones y perspectivas

En este trabajo se han presentado experimentos relacionados con la extracción de eventos indicadores que utilizan un número fraccionario (tomando como base el 100) para expresar una unidad de valor.

El trabajo aporta una serie de patrones morfosintácticos útiles en la tarea de identificación de eventos indicadores. Se han extraído y presentado aquellos patrones lingüísticos que han mostrado una mayor regularidad de ocurrencia en los eventos indicadores.

Como trabajo a futuro se considera incrementar sustancialmente el número de noticias sobre el cual se llevarán a cabo los experimentos y llevando a cabo una evaluación manual de todos y cada uno de los eventos extraídos manualmente, lo cual será por supuesto una tarea costosa desde el punto de vista del tiempo y esfuerzo humano.

Es importante analizar el conjunto inicial de eventos indicadores, a fin de poder enriquecer los patrones morfosintácticos y encontrar otros que aunque poco frecuentes, sean de interés en la extracción de la información basada en eventos.

Referencias

1. Andersen M., Hayes J., Huettner A., Schmandt L., Nirenburg I.: Automatic Extraction of Facts from Press Releases to Generate News Stories, Proceedings of the Third Conference on Applied Natural Language Processing, Association for Computational Linguistics, ANLC 9̀2, pp. 170–177 (1992)
2. Gil-Vallejo L., Castellón I., Coll-Florit M.: Hacia una definición de la similitud verbal para la extracción de eventos, Centro Virtual Cervantes (2015)
3. Hernández L. D.: Sistema web para identificar eventos y actores en textos periodísticos, royecto terminal, Departamento de Sistemas, Universidad Autónoma Metropolitana Unidad Azcapotzalco, México (2015)
4. Martín F. J., Ruiz J. L.: Procesamiento del lenguaje natural, España: Universidad de Sevilla. Disponible en: https://www.cs.us.es/cursos/ia2/temas/tema-06.pdf (2013)
5. Moncecchi G., Rosá A.: Reconocimiento automático de eventos en textos, Proyecto de grado, Facultad de Ingeniería, Universidad de la República, Uruguay (2010)
6. Padró L., Stanilovsky L.: FreeLing 3.0: Towards Wider Multilinguality Proceedings of the Language Resources and Evaluation Conference (LREC 2012) ELRA.Istanbul, Turkey (2012)
7. Priego-Sánchez B., Pinto D.: Identification of Verbal Phraseological Units in Mexican News Stories. Computación y Sistemas, Vol 19(4), pp. 713–720 (2015)
8. Ramos O., Pinto D., Priego-Sánchez B., Olmos I., Beltrán B.: Análisis empírico de la dispersión del español mexicano. Research in Computing Science 74, pp. 9–19 (2014)
9. Real Academia Española.: Diccionario de la lengua española [Dictionary of the Spanish Language] (22nd ed.). Madrid, Spain (2001)
10. Steinwart I., Christmann A.: Support Vector Machines (1st ed.). Springer Publishing Company, Incorporated (2208)
11. Sutton C., McCallum A.: An introduction to conditional random fields for relational learning, in: L. Getoor, B. Taskar (Eds.), Introduction to Statistical Relational Learning, Ch.1, MIT Press (2007)
12. TES (Terminology Extraction Suite): Distibución para Windows—Traducció, Traduccio.blogs.uoc.edu, urlhttp://traduccio.blogs.uoc.edu/2012/04/13/52/
13. TreeTagger, urlhttp://www.cis.uni-muenchen.de/ schmid/tools/TreeTagger/
14. Wonsever D., Malcuori M., Rosá-Furman A.: SIBILA: Esquema de anotación de eventos". Reportes Técnicos 08-11. UR. FI INCO (2008)

ISSN 1870-4069

Comparison between LSP and MFCC parameterizations in a Spanish Speech Synthesis System

Carlos Franco[1], Abel Herrera[2], Boris Escalante[2], Fernando Del Río[2]

[1] Benemérita Universidad Autónoma de Puebla, Facultad de Artes, Puebla, Mexico

[2] Universidad Nacional Autónoma de México, Laboratorio de Tecnologías del Lenguaje, Mexico

francocarlos@gmail.com, abelhc@hotmail.com, boris@unam.mx,haitosan@hotmail.com

Abstract. Voice parameterization using Line Spectral Pair was implemented to a Mexican Spanish HMM-based Speech Synthesis System. Five phrases were synthesized and statistically validated by applying a MOS test to 30 listeners who analyzed the original voices versus synthetic voice. Results were compared with a synthesizer where MFCC was used as voice parameterization. Two aspects were evaluated on the voice: Naturalness and Intelligibility. The comparison shows that LSP parameterization is above the mean score and pointed better than MFCC.

Keywords. Speech synthesis, line spectral pair, MFCC, Spanish language synthesis.

1 Introduction

By the end of the 20th century, the Festival speech synthesis [1] system together with its variants, CLUNITS and CLUSTERGEN reached a remarkable naturalness in artificial speech. Festival-CLUNITS parameterizes sub-phonemes, it belongs to a kind of synthesis named parametric speech synthesis (SPSS). Festival-CLUSTERGEN [2] on the other hand, works with acoustic sub-phonemes it belongs to the unit selection speech synthesis types. Both Festival programs are concatenative synthesizers.

Some authors in this paper replicated these achievements by adjusting Festival to central Mexico Spanish [3]. Hidden Markov Models (HMM) were applied in speech synthesizers to search units using stochastic methods which meant some advancement in how natural the Synthesizer sounded. Such systems are known as Hidden Markov Models as Text to Speech Synthesis (HTS) [4] these are SPSS, in other words, they use parameterized speech.

Still today, the doubt remains wether the HTS systems overcome the Festival-CLUSTERGEN system [5]. Some of the authors adapted an HTS system to central Mexico spanish which showed certain improvements compared to Festival [3].

pp. 19–27; rec. 2017-09-05; acc. 2017-10-20

In Festival CLUNITS and HTS, the preferred parameterization was MFCC [6]. For the latter, the parameterization together with pitch and duration conform the speech signal, sometimes delta and delta-delta are included [7]. Unfortunately, no substantial improvement was shown.

Another essential aspect of HTS is the inclusion of a Vocoder filter to recreate the speech signal. This process occurs having pitch, duration and parameterization as inputs [6]. Unlike Festival-CLUSTERGEN where instead of synthesizing through a Vocoder, acoustical units are concatenated and smoothen out.

During the first decade of the current century, another mildly successful parameterization was experimented with. It was known as STRAIGHT. It was also validated in a Mexican spanish system. [8].

The success of MFCC and LSP was not due to their compression in terms of voice segments but because such parameterizations are solidly based on the acoustical characteristics of the human voice.

When LPC was first created, several variants of it came along. Among those Line Spectral Pair was very well received. LSP take into account the acoustical behavior of the speech signal within the vocal tract. At that point in time, this feature was not relevant, and the parameterization was no longer worked on for some time.

It has not been as widely used as MFCC, Nakatani and colleagues [9] hypothesized that MF-LSP is a little more efficient than LSP. The authors decided to continue that line of research with its respective experiments. After adjusting and statistically validating the system. The authors conclude that it efficiently produces speech synthesis in Spanish Language using LSP. Its naturalness and intelligibility were qualified above the mean and above previously validated MFCC based synthesis.

It has just happened during the last four years that major changes took place in Speech synthesis. From 2013, Deep Neural Networks [10] are used to synthesize speech. Different network architectures report improvements in the voice quality [11]. A major application of the current work is in hybrid systems where LSP are used in applications for low memory devices [12].

This document is organized as follows: Section 2 mentions related efforts towards validating LSP as a speech synthesis parameterization. Section 3 three briefs the reader on HTS Speech Synthesis. Section 4 summarizes the theory behind the parameterization. Section 5 describes LSP as speech parameterization. The experiments and its results are described in section 6 and the conclusions are given in section 7.

2 Related Work

LSP parameterization of a speech signal has been in the interest of speech synthesis and recognition for the last three decades. Nakatani [9] and colleagues evaluated LSP parameterized phrases, but their study was exclusively focused on analyzing isolated phonemes in japanese and not entire phrases. Arakawa and colleagues [13] applied LSP to improve certain features of the STRAIGHT synthesis system, but the principles of such system differ from those of the system the authors experimented with. Bäckstörm in his doctoral project [14], [15] makes a complete mathematical analysis of LSP but his work is theoretical and not focused exclusively on speech signals. Tokuda and his team [4] left the door open to experiment with Either LSP or MFCC but they focused on the HTS (Hidden Markov Models as Text to Speech Sythesis) system from a global perspective and do not report results on speech parameterization effectiveness.

3 Speech Synthesis Using HTS

HTS (Hidden Markov Models as Text to Speech Sythesis) is a proposal from the 2000´s. This type of synthesis decomposes a voice signal in three vectors which include its three main features: Mel General Cepstral coefficients MGC [16], F0 and duration. In practice, these vectors are obtained with a software named Signal Processing Tool Kit SPTK [4].

The vectors are accessed non-linearly to obtain the correct phoneme sequence in a spoken phrase. Therefore, the stochastic selection algorithm of Hidden Markov Models HMM is used in contrast with other synthesis systems, such as Festival [17] were phonemes are selected using a linear method named CART [18].

To compute the probability of the HMMs, the creators of HTS took advantage of a free distributed system developed by the university of Cambridge. The program is known as Hidden Markov Model Toolkit HTK [19].

HTK was originally designed for speech recognition.

Figure 1 shows a general scheme of HTS. More details can be found on the references [5] and the HTS website [4].

Before being able to synthesize a phrase, HTS need to be trained with the desired language specifications. Other characteristics are as well defined in the training stage (e.g. parameterization, number of coefficients, sampling frequency, etc.)

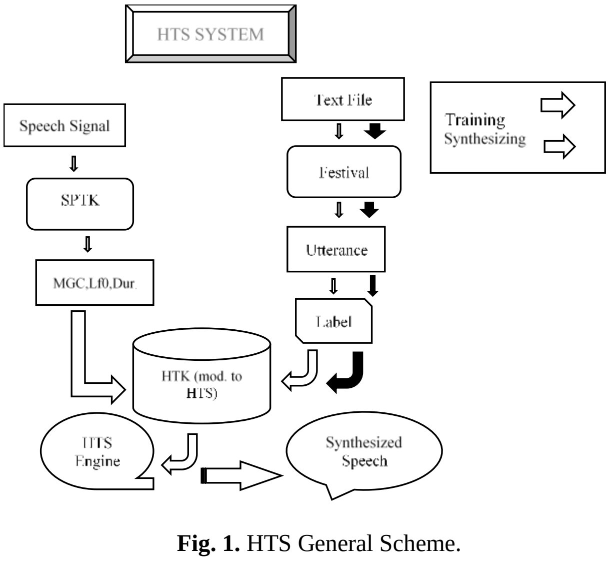


Fig. 1. HTS General Scheme.

The system is trained by inputting 300 audio files containing the recording of phonetically balanced phrases and text files with their respective transcription. The highest probabilities of occurrence of a phoneme sequence will be calculated within the HMMs to obtain the better combination. Text to phoneme conversion is done through Festival [17]. Since Festival was originally designed for english language synthesis, when a different language is used, the system must be adapted to process the grammatical features of such a language. All these grammatical features are coded in a software called lexicon. A lexicon in spanish indicates Festival the use of stressed vocals, letter “ñ”, differences between phonemes like /c/ or /z/ among others. The current system uses a lexicon created originally for Andalusian spanish named Junta de Andalucía. It was chosen because iberic spanish is grammatically identical to mexican Spanish, no further modifications were needed. Except for substituting “c” and “z” letters for an “s” when the desired synthesized phrase is being written. Text to phoneme conversion is performed in the following order: Sentence to phrase, phrase to word, word to syllable and syllable to phoneme [2].

Once the conversion is finished, Festival delivers a utterance (.utt) file. The actual synthesis process takes place in a software named HTS Engine, utterance files must be reorganized to be compatible with it. For that purpose, they are changed into label (.lab) files.

Input data to the system were used before in the spanish synthesis MFCC parameterization training. Such data consists of 300 phrases recorded as wave files in an anechoic chamber by a male professional radio speaker. The wave files were coded into RAW files which contain the same information of the wave file except for a header.

The other input data simultaneously processed are the label (.lab) files. These are text files which indicate HTS Engine the desired phoneme sequence (e.g. sentence, phrase, word, syllable) of the phrase to be synthesized.

The RAW files are decomposed in three vectors: One vector contains Mel General Cepstral Coefficients; the second vector contains the phrase LogF0 and the third one the phrase duration. These three elements are stored in a three-streamed HMM which is in practice a Gaussian matrix. Their delta and double delta Coefficients are also considered to smooth out the wave transitions within each other. A common practice in speech processing. This model is named hmm0. The calculations are done based on a previously given phoneme probability master label file MLF [19].

The model hmm0 should be divided into smaller models to separate the different phoneme values. For that matter, the mean of hmm0 is calculated generating a new three-streamed model named hmm1. The probabilities stated in the MLF are then condensed in a Master Macro File MMF. Based on this file probabilities, the process is repeated iteratively until several HMM models are formed. The number of HMM models is previously defined by the user.

Once the HMM models are completed, their probabilities are computed following a Viterbi algorithm and grouped into single phoneme gaussians. Thus, for example, all the /a/ phonemes are together in a same group. And the selection process will be linear.

The synthesis takes place in a piece of software named HTS Engine [5] which is a vocoder filter driven by two sound sources: Sinusoidal for voiced sounds and white noise for unvoiced sounds. The formers emulate those voice sounds produced by the vocal cord vibrations and the others are phonemes produced by air currents passing from the lungs to the mouth. The filter frequencies correspond to those of the phonemes that will be produced.

4 Mel General Cepstral

The concept of Mel General Cepstral MGC [16] includes two different voice parameterizations: Mel-Cepstral Analysis and Linear Predictive Coding LPC.

The Mel-Cepstral analysis is quite popular. It was the first effort of the authors when dealing with HTS based speech synthesis. The part which corresponds to LPC is the starting point to the authors proposal using Line Spectral Pair LSP.

Mel General Cepstral parts form the speech signal spectrum H(z) defined as follows:

$$H(z) = S_{\gamma}^{-1}\left(\textstyle\sum_{P=1}^{N} A_P z^{-P}\right), \qquad (1)$$

where Sγ is a generalization of the logarithmic function:

$$S_{\gamma} = \begin{cases} \frac{\omega^{\gamma}-1}{\gamma}, \ 0 < |\gamma| \leq 1 \\ \log\omega, \ \gamma = 0 \end{cases} . \qquad (2)$$

Applying this principle to H(z) in equation (1) provides the following information:

$$H(z) = \begin{cases} (1 + \gamma \sum_{P=1}^{N} A_P z^{-P})^{\frac{1}{\gamma}}, 0 < |\gamma| \leq 1 \\ \exp \sum_{P=1}^{N} A_P z^{-P}, \gamma = 0 \end{cases} , \qquad (3)$$

when γ=0, the speech parameterization corresponds to the Cepstrum definition, in which MFCC parameterization is based on. On the other hand, if γ=1 LPC parameterization is obtained.

To convert from LPC to LSP, we define the filter $H(z) = 1 + \sum_{P=1}^{N} A_P z^{-P}$ as the sum of two polynomials P(z) and Q(z) [20] each of them is defined as:

$$P(z) = A_p(z) - z^{-(p+1)} A_p(z^{-1}), \qquad (4)$$

$$P(z) = 1 + \textstyle\sum_{p=1}^{P} (a_{p+} a_{P+1-p}) z^{-p} + z^{-(p+1)}, \quad (5)$$

$$Q(z) = A_p(z) + z^{-(p+1)} A_p(z^{-1}), \qquad (6)$$

$$Q(z) = 1 + \textstyle\sum_{p=1}^{P} (a_{p-} a_{P+1-p}) z^{-p} - z^{-(p+1)}. \qquad (7)$$

Every polynomial has P/2 pairs of complex conjugate roots for this reason, the above written equations can be represented the following way:

$$P(z) = (1 + z^{-1}) \prod_{i=1}^{\frac{P}{2}} (1 - z^{-1} e^{-j\omega})\,(1 - z^{-1} e^{-j\omega})$$

$$= (1 + z^{-1}) \prod_{i=1}^{\frac{P}{2}} (1 - 2cos\omega_i z^{-1} + z^{-2}), \qquad (8)$$

$$Q(z) = (1 - z^{-1}) \prod_{i=1}^{\frac{P}{2}} (1 - z^{-1}e^{-j\theta})\,(1 - z^{-1}e^{-j\theta})$$

$$= (1 - z^{-1}) \prod_{i=1}^{\frac{P}{2}} (1 - 2cos\theta_i z^{-1} + z^{-2}). \qquad (9)$$

The values of ω and Θ represent respectively in P(z) y Q(z) the formant frequencies of the format to synthesize. All of them contained in (0, π) they are known as Line Spectral Frequencies LSF.

5 Using LSP to Synthesize a Speech Signal

The authors decided to test this type of parameterization and adapt it to the current HTS Spanish system. The system by default decomposes the speech signal in Mel General Cepstral Coefficients. It is based on a mathematical concept that unifies MFCC and LSP based on the equation (3) mentioned in the previous section.

This process takes place using SPTK. A technical manual with coding details will be published by the authors.

The authors decided to test it for several reasons: First, LSP is based on Linear Predictive Coding (LPC) which parts form seeing the human vocal tract seen as a filter and the formant frequencies as the filter coefficients. The spectra obtained based on vocal tract models tend to resemble natural speech remarkably. Even more, LSP takes into account more data than LPC which results in a richer quantization of the original speech signal. An LSP voice filter is more stable in nature, the mathematical demonstration can be found in [15]. The size of the audio files is smaller than that of the files using MFCC. Finally, and most important: There are little documented on Speech synthesis using LSP and particularly in spanish, no documentation was found.

6 Evaluation and Comparison of Both Parameterizations

With the purpose of verifying the quality of the synthesized voiced, the authors tested both parameterization techniques: LSP and MFCC [8]. Two aspects were validated: Naturalness and Intelligibility. For naturalness, a MUSHRA Test was performed. To validate intelligibility, five phrases were played to an audience who had to write them down.

6.1 MUSHRA Test

The MUSHRA test [21] is a standard validation test recommended by the International Telecommunications Union. It was specifically designed for the evaluation of different audio codecs. It is organized in a way that the listener analyzes the same audio content codified in different forms including the original recording without compression and the original recording low-passed filtered to anchor the listener to the reference.

A population of 30 listeners was surveyed. All the listeners were either audio specialists or music technology students, since the ITU recommendation requests for experimented listeners. Each person listened to 5 phrases in five different versions: The Original Recording, The Original recording passed through a 100 Hz cut-off filter, a synthesized phrase using MFCC parameterization and a synthesized phrased with LSP parameterization. The subject sat in front of a computer and listened to the phrases through headphones with a SNR of 93 dB. Two aspects on the audio were validated, every phrase had to be qualified by the listener on a 0 to 100 scale according to the norm. At least one phrase had to be qualified with 100.

6.2 Intelligibility Test

Usually, intelligibility tests in speech coding, are focused on proving how easy is for a listener to understand a phrase when the speech signal is masked or filtered. In this case, the interest of the authors was to evaluate how easy the synthetic phrase was to understand depending on the parameterization used.

The best way to validate intelligibility is by dictation. Five LSP and five MFCC synthesized phrases were played to a group of 27 listeners who had to write them down. The listeners written dictations were the individually marked. The given marks to each phrase were correct or incorrect. The MFCC phrases averaged a score of 0.84 whereas the LSP phrases averaged 0.89.

Table 1 shows the obtained mean scores for both aspects. Note that the original reference and its anchor were only used to measure naturalness.

7 Conclusions

As we could learn from the results regarding naturalness and intelligibility - shown in Table I, there is an improvement in both aspects when LSP is chosen as voice parameterization. In terms of file size, LSP speech parameterization files are smaller than MFCC parameterization files. This reduction can be important in terms of data transferring and data storing economization.

The authors consider LSP speech parameterization as a new standard in future works related to speech synthesis in Laboratorio de Tecnologías del Lenguaje FI UNAM.

MFCC parameterization on the other hand is not much below LSP in qualifications. It is widely used in several recognition and synthesis systems. It will be hardly replaced by a speech parameterization which is only a few points ahead in acceptance.

The anchor in the MUSHRA test is used precisely to unconsciously remind the listener what the reference was. Surprisingly it was marked below both parameterizations.

The reference was unmistakably identified by all the listeners. This condition is a reminder that a synthesizer that sounds as natural as a human is still a relevant challenge in the field.

After conducting the experiments described in this document, to new voices were developed using male and female speakers. Both were parameterized with LSP. They have not been statistically validated but early tests showed certain success in intelligibility and naturalness in the authors opinion, their validation remains for future work. Experimenting with different speakers to be synthesized would shed certain light in determining which features should a human voice have to serve as a model for a synthetic voice.

Table 1. Evaluation Results.

Type	Naturalness	Intelligibility
Reference	100	N/A
Anchor	62.6	N/A
LSP Parameter	69.5	0.89
MFCC Parameter	61.4	0.84

Some of the possible failures in imitating human speech are related to the way the phonemes are chosen and concatenated. Adjustments in that stage may lead to an improvement in quality independently of the chosen parameterization.

Current studies on speech synthesis and recognition are walking away from HMM and searching the use of Deep Neural Networks DNN as the new phoneme selection system

Acknowledgements. The current work was sponsored by a CONACYT scholarship and PAPIIT IT 102314 of DGAPA-UNAM. The authors appreciate the suggestions of Dr Felipe Orduña regarding the statistical methods of evaluation.

References

1. Taylor, P. , Black, A.W., Caley R.: The Architecture of the Festival Speech Synthesis System. Proc. 3rd ESCA Work, Speech Synth, 147–151 (1995)
2. Black, A.: CLUSTERGEN: a statistical parametric synthesizer using trajectory modeling. INTERSPEECH (2006)
3. Herrera-Camacho, A., Del Río-Ávila, F.: Development of a Mexican Spanish Synthetic Voice Using Synthesizer Modules of Festival Speech and HTSStraight. Int. J. Comput. Electr. Eng, 36–39 (2013)
4. Tokuda, K., Nankaku, Y., Toda, T., Zen, H., Yamagishi, J., and Oura, K.: Speech Synthesis Based on Hidden Markov Models. Proc. IEEE, 101(5), 1234–1252 (2013)

5. Tokuda, K., Zen, H., Black, A.: An HMM-based speech synthesis system applied to English. IEEE Speech Synth. Work (2002)
6. Zen, H., Tokuda, K., Black, A.: Statistical parametric speech synthesis. Speech Commun. 1229–1232 (2002)
7. Tokuda, K., Kobayashi, T., Imai, S.: Speech parameter generation from HMM using dynamic features. 1995 Int. Conf. Acoust. Speech, Signal Process, 1(5), 660–663 (1995)
8. Franco, C., Del Rio, F., Herrera, A.: ATINER Conference Paper Series Speech Synthesis of Central Mexico Spanish using Hidden Markov Models. 1–12 (2016)
9. Nakatani, N., Yamamoto, K., Matsumoto, H.: Mel-LSP Parameterization for HMM-based Speech Synthesis. Eurasip Proc. SPECOM 2006 (2006)
10. Lu, H., King, S., Watts, O.: Combining a Vector Space Representation of Linguistic Context with a Deep Neural Network for Text-To-Speech Synthesis (2014)
11. Qian, Y., Yuchen, F., Wenping, H., Soong, F. K.: On the Training Aspects of Deep Neural Network (DNN) For Parametric Synthesis. Microsoft Reasearch (2014)
12. Soong, F., Juang, B.: Line spectrum pair (LSP) and speech data compression. In: ICASSP ’84. IEEE Int. Conf. Acoust. Speech, Signal Process, 9(9), 37–40 (2013)
13. Arakawa, A., Uchimura, Y., Banno, H., Itakura, F., Kawahara, H.: High quality voice manipulation method based on the vocal tract area function obtained from sub-band LSP of STRAIGHT spectrum. ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process - Proc, 2, 4834–4837 (2010)
14. Bäckström, T., Magi, C.: Properties of line spectrum pair polynomials-A review. Signal Processing, 86(11), 3286–3298 (2006)
15. Backstrom, T.: Linear predictive modelling of speech - constraints and line spectrum pair decomposition. Matrix (2004)
16. Tokuda, K., Kobayashi, T., Masuko, T., Imai, S.: Mel Generalized Cepstral Analysis - A Unified Approach to Speeh Spectral Estimation (1994)
17. Taylor, P., Black, A., Caley, R.: The architecture of the Festival speech synthesis system (1998)
18. Black, A., Taylor, P.: Automatically clustering similar units for unit selection in speech synthesis (1997)
19. Young, S.: The HTK Book. J. Chem. Inf. Model. 53(9), 1689–1699 (2013)
20. Zheng, F., Song, Z., Li, L., Yu, W., Wu, W.: The distance measure for line spectrum pairs applied to speech recognition. In: Proc. 5th Int. Conf. Spok. Lang. Process, 1998 (ICSLP ’98), 1123–1126 (1998)
21. Itu-BS.1534: Method for the subjective assessment of intermediate quality level of audio systems Policy on Intellectual Property Right (IPR). Series of ITU-R Recommendations, 1534–3 (2015)

ISSN 1870-4069

A Document Profile for Improving Information Retrieval Systems

Antonio Guillén, Yoan Gutiérrez, Rafael Muñoz

University of Alicante, San Vicente del Raspeig,
Department of Software and Computing Systems, Alicante, Spain

{aguillen,ygutierrez,rafael}@dlsi.ua.es

https://www.dlsi.ua.es

Abstract. In recent year the great popularity that enjoys mobile technologies has led most users to become consumers and producers of information on the network. Many studies speak about this phenomenon as an activity that is capable of doubling or tripling existing content on an annual basis. The huge amount of information makes the current user oriented systems, like retrieval systems, recommender systems and others, become less efficient, especially when users require specific information and answers according to their needs and preferences. These facts make necessary to equip these systems with proper Natural Language technologies able to provide the information that users demand adapted to each context and content type. In this article, it is presented a study of some Natural Language Processing technologies that can be useful to facilitate the proper identification of documents according to the user needs. For this purpose, it is designed a document profile that will be able to represent semantic meta-data extracted from documents. The research is basically focused on the study of different language technologies in order to support the creation this novel document profile proposal from semantic perspectives.

Keywords. Retrieval system, NLP technologies, document profile, meta-data, web searching.

1 Introduction

The Internet provides large amounts of information through many types of documents. Users require an easy way to filter these documents to find out the most appropriate documents for their interests, capabilities and needs. These documents also can be retrieved by other entities for market studies, classify and index documents, detect fake information or illegal activities on the Web. These aspects can be treated by the study of NLP areas, tasks, methods and tools.

In this paper, is presented a novel document profile proposal for improving information retrieval systems. Also, is presented a study to determine which NLP technologies are the most suitable for this purpose. In this study, is addressed which technologies are available currently, estimate its automation and reliability degree, the problems that can be found on applying them and the most appropriate document's type. Also, is described this novel document profile proposal and is presented a selection of documents to be treated in the proposal. A document can be defined as a short or long unity of information, principally obtained from websites (user posts, press articles, comments, reviews, etc.). The aim of this study is to investigate the different features that can be extracted by means NLP technologies able to provide enough information for setting up useful meta-data. The main purpose of this meta-data is improving retrieval systems and search engines to get better search results. Also, another important purpose is to improve recommender systems through common meta-data between documents.

The paper is organised as follows: Section 2 shows the related work of this research. Section 3 describes the novel document profile proposal. Section 4 presents an example of document profile with a real document and a possible usage. Section 5 addresses a study of selected NLP tasks and technologies which serve for supporting the proposal. Section 6 exposes the conclusions and future works.

2 Related Work

Two areas closely related to Document Profiling are User Profiling [1] and Recommender Systems [2]. Although those research areas differ from Document Profiling we can find some features useful to be reused and incorporated into this work. For instance, author identification methods help to reveal personal information (i.e. age, gender) about document's author. In the case of reusing Recommender Systems research technologies, there are interesting researches for our work. Paper [3] propose an approach for constructing user profiles based on the interaction and user behaviour on the Internet for news recommendation, this profile is relevant to our work because is focused on news documents and includes information about time expressions, location, and topic tags.

There are other works related to the document profile proposal. Paper [4] justifies the importance of this proposal for Information Retrieval. This work emphasises the search quality by using meta-data information. Paper [5] uses the term *document profile* to refer to N-Gram frequencies for text categorisation. Paper [6] explains a review analyser that extracts product reviews and analyse its information. As result sentiment polarity classifications are extracted. Other related works are focused on improving retrieval systems [7] tough efficient Information Retrieval techniques and algorithms. It is interesting because it reveals some useful web searching techniques and algorithms.

3 Document Profile Proposal

The proposal mainly consists of designing a document profile able to represent different features extracted once NLP technologies are applied on documents, assisted or not by humans. As can be seen, there are many features that can only be extracted automatically from documents by using NLP technologies. This work pretends to unify most of these NLP technologies as a whole able to characterise documents from different points of view.

3.1 Document Selection

As a first approach, this work will be focused on English documents from the Internet. This is due to NLP technologies have been mostly developed to cover this language and which makes easier to find out NLP tools. Nevertheless, the overall strategy is still valid for other languages. The principal documents addressed are those mostly available on the Internet[1]. Those are press media, social posts, product/service reviews, blogs or personal websites, academic/science papers, tutorials or instruction manuals, etc. The complete documents' schema and meta-data properties are shown in this source[2]. Is defined two classes of documents (leaf and conceptual), leaf document represents concrete documents that can be obtained the profile, and the conceptual document is defined as group documents that share common features.

Leaf documents represent Internet content such as blogs, tweets, news, etc. This work is focused on these documents because is daily used by users on the Internet, and considered in many NLP corpora.

Next, is described some of these documents, which can be represented by the leafs:

Press Document: A press document consists of a digital press document provided by any press media (TV, written, etc.). Usually, this document is a large text which use to include information about entities like dates, persons, places, institutions, etc. Some examples of press documents are on-line versions of newspapers such as The New York Times; The Times; The Washington Post; The Economist; or news channels like CNN, BBC World News etc.

Social Post: A social post is a content coming from any social network or forum. Usually, it is a short informal text written by an Internet user. Some examples of social post documents can be found in popular social networks[1] like Twitter, Facebook, Instagram, etc.

Product/Service Review: A product review is an opinion expressed by a consumer about a product, service, video, etc. Usually, it is a short informal text. Some examples of product reviews can be found on e-Commerce websites like Amazon[3], video-sharing websites like Youtube, etc.

Blog: A blog document is an informal diary-style text entry about an author ideas. Can be a large or short informal text written by a user. Many examples of blogs are found on the Internet about many subjects or topics. Some blogs[4] examples are The Huffington Post, TMZ, Business Insider, etc.

[1] http://www.quantcast.com/top-sites

[2] http://ow.ly/pTaR30dWTj7

[3] https://www.amazon.com/review/hall-of-fame

[4] http://www.ebizmba.com/articles/blogs

Personal Web: A personal web document is a personal website or biography in which a person describes synopses of her/his life. Usually, it is a large formal text about a person in which it can be identified many clues about her/his life like dates, places, companies, other persons, etc. Also, it can be considered a personal web a general purpose webs[1] like commercial websites, corporations, brands, products or services.

Academic Document: An academic document is any digital content used to the learning process, i.e. digital books, slides, etc. Usually, it is a large formal text used for whatever level teaching.

Instruction Manual: An instruction manual is a large formal text for describing technical procedures or how instructions for using products or services.

Tutorial: A tutorial document is a set of instructions to learn some task. Usually, it is a large informal text written by users.

Scientific Document: A scientific document is whatever scientific environment material like articles, slide-shows, etc. Usually, it is a large formal text used for whatever level teaching.

Literary Document: A literary document is a text content usually represented by E-Books. In this work, it will be considered only a portion of a literary work because a complete work maybe is very large to analyse. Therefore, a literary document represents a short formal text written by an author.

Technical Document: A technical document represents textual contents that describe professional or commercial procedures (corporate websites, descriptions of products or services), medicine and health (medical prescriptions, pharmaceutical leaflets), public administration (public calls, court judgements, tenders), among others.

3.2 NLP Tasks Selection

In this research, a selection of NLP tasks[5] for designing the profiling meta-data properties is carried out. This list has been taken into account to be included in this novel document profile proposal. This list has been taken into consideration to be included in this novel document profile proposal. At this way, it can be obtained a long quantity of meta-data contributing to filling document profiles. As can be seen, is chosen some of the most active NLP tasks nowadays. So, the main goal at this stage is to find out friendly access NLP technologies (i.e. tools, APIs, demos, etc) to test and simulate this proposal in a real scenario. At this way, it would be demonstrated the viability of this proposal, not just providing a document profiling scheme.

3.3 Meta-data Properties

The meta-data properties defined for our proposal can be found in the following source[6]. Notice, that each of these properties indicates the NLP technology acronym from which it is obtained. Is included a brief description of each meta-data property.

[5] http://ow.ly/MPO130e24Jr

[6] http://ow.ly/tQos30dNVcS

3.4 Document Profiling Algorithm

Is defined an algorithm to generate document profiles using the available NLP technologies. Algorithm 1 shows the steps to generate a profile from a Web document.

Algorithm 1: Document Profiling

```
program GenerateProfile
   {Require: url};
   begin

     cont := getContent(url);
     sum := getSummary(cont);
     type := getType(cont);
     profile := newProfile(cont,sum,type);
     listProperties := getProperties(type);
     foreach (prop in listProperties) do
       nlpTech := prop.getNlpTech();
       value := nlpTech(cont,summ);
       profile.add(prop,value);
     endfor

   end.
```

The steps are specified as follows: Web document content is extracted from url. A short version of the document using the summarising technology is generated. Detection of the document type. In the first approach, this detection will be manual. It is intended to do a classification system for auto-detection using the document's schema proposed. Generate an initial profile from scratch, only including the meta-data obtained currently (i.e. content, summary and type). Get the specific meta-data properties for this document type. Iterate on meta-data property list. Get related NLP technology of meta-data property. Invoke NLP technology with the content or summarised document (depends if require a large or short version document). Add result value obtained from NLP technology to the profile.

4 Document Profile Example and Usage

In order to figure out how would be a document profile, we have prepared an example considering a document taken from the CNN website. The document is a real news article about Everest's climber George Mallory. He's the first person who tried climb to summit the Everest. He was disappeared and his body was found 75 years later. Expert people tried to discover if he reached the top of Mount Everest or not. The complete news article it's available in CNN website[7]. The procedure follows as to describe the algorithm 1, obtaining the next document profile[8].

A brief description of the features set is commented following. (i) The identification *001* is auto-generated. (ii) The document type detected is *Press Document.* (iii) *Content*, *Title* and *Date* meta-data properties can be extracted directly from the article, and the source corresponds to the web URL of article. (iv) The content summary is generated. (v) The topics list represents the most frequently used terms in the text. (vi) The region is obtained from the text, in this case, it talks about the north ridge of Mount Everest located in Tibet (China). (vii) About the subject areas detected can see *History* because it is a historical news article and *Sports* it talks about climbing. (viii) The language detected is English. (ix) The article does not present any rating. (x) Keywords are the more representative words and expressions of the whole text. (xi) In this article, it is not detected any ideological orientation because it only talks about a historical fact related to sports. (xii) Sentiment polarity determines that it is a positive news for people interested in history, climbing and the Mount Everest. (xiii) It is detected the emotion category *Surprise* because the man's body has been found 75 years later, something unexpected. (xiv) Some time expressions are detected in the text, these expressions have been converted into date format. (xv) The name entities detected refer persons and locations retrieved in the text. (xvi) In this article is not detected irony. (xvii) Reading complexity is *Easy*, with *Normal* textual formality and *Neutral* writing formality because the news article is from a serious press media. (xviii) The article is feasible to be read by 16 years old people over, since it is a neutral and simplified information. (xix) The age of author is predicted in the ranges of 13-17 years old or 65-100 years old. (xx) The gender is predicted as *Female.* (xxi) The press type is classified as *News Article* and its veracity is *Truthful*, taking into account the formality of press media.

One interesting use of our proposal is shown in this source[9]. Through document profile, it is possible to get related documents with common meta-data. For instance, with some meta-data could navigate among different documents that can be interesting to the user. This usage provides a great experience of knowing automatically which are the documents more appropriate to each user.

5 NLP Technologies Study

Each NLP task considered is described in this work and exposes some available technologies: tools, APIs or demos related. In some cases, only methods and algorithms had been found. Mainly works with evaluations and results are presented. In this study, the interests are mainly focused on available technology (Web service/API, programming library) with certain automation and reliability degree to be able to be incorporated in future frameworks or prototypes. The results of this study are exposed in a comparison table.

[7] http://ow.ly/7JtB305V8z6

[8] http://ow.ly/Pnmj30dNVtQ

[9] http://ow.ly/Etos30gfoH9

Text Classification: Text classification task (TC) consists of identifying the type of document by content analysis. The relevance of this task in document profiling is to determine which type of document is analysed, depending its type different features would be extracted. For example, Dandelion API[10] categorises plain text on eight categories: business, economy, sports, etc. In this approach, the type is considered a precondition data to generate a profile, because it doesn't exist a TC technology that uses the concrete leaf document types proposed documents' schema.

Information Extraction: Information Extraction task (IE) has been addressed by the paper [8] as a way to search and obtain text on large volumes of unstructured information to filter relevant information, using regular expressions, rules and patterns. This is useful for document profiling because many complex meta-data can be obtained directly from the document. Since this work is mostly focused on Web documents there are too many tools like DEiXTo[11] that can extract information from the W3C DOM documents. The problem with these tools is the low-automation degree due to they should be re-configured.

Topic Recognition: Topic Recognition task (TR) consists of identifying topics in the text. This task is interesting to classify a document in multiple categories or topics and know the different aspects that dealt the document. TextRazor[12] is a Web tool that lists topics from a text. Also, another Web demo is Meaning Cloud[13] that offers many NLP services among which topic recognition is included.

Keyword Extraction: Keyword Extraction task (KE) is the automatic extraction of relevant terms from a document. Unlike TR, KE doesn't intend to know the different aspects that dealt the document, KE extracts terms that best describe the subject of the document. Statistical Keyword Extraction Tool (SKET) [9] is a programming library for extracting keywords from the text.

Named Entity Recognition: Named Entity Recognition task (NER) tries to locate and classify named entities according to different categories like names of persons, organizations, etc. Stanford NER [10] is a NLP technology available as a programming library and evaluated in some scenarios, domains and corpora, that offers useful NER services.

Time Expression Recognition: Time Expression Recognition task (TER) consists of obtaining temporal expressions from texts. This information is useful to extract historical facts in texts or documents. Stanford SUTime [11] is a tested technology and available to be used as a programming library. An alternative is TIPSem [12] which has been tested and available as API.

Automatic Summarization: Automatic Summarization task (AS) obtains a reduced text from a larger text content. It's interesting to obtain a short

[10] http://dandelion.eu

[11] http://dexi.io

[12] http://www.textrazor.com

[13] http://www.meaningcloud.com

version of the same document. Paper [13] presents a summarization system for various purposes and domains. This system has been evaluated and is available as API.

Domain Detection: Domain Detection task (DD) is part of Semantic Parsing. This task detects the meaning of sentence using probabilistic semantic models. Paper [14] is focused on ISR-WN which is able to detect domains or categories from different resources obtaining and using relevant semantic trees from a text.

Language Identification: Language Identification task (LI) consists of identifying the language. AlchemyLanguage[14] is a Web demo that offers many NLP services for that purpose.

Polarity Classification: Polarity Classification task (PC) is part of Sentiment Analysis. According to paper [15], this consists of determining whether a text in positive, negative or neutral. Many PC approaches are using machine and deep learning which obtains good results. Among others, are mentioned two relevant works: [16] which presents an approach using supervised statistical machine learning, and Stanford Sentiment [17] that is available as a programming library.

Emotion Detection: Emotion Detection task (ED) is part of Sentiment Analysis. This consists of identifying some emotions expressed in texts. The following set of the basic emotional categories proposed by Ekman [18]: Anger, Disgust, Fear, Joy, Sadness and Surprise. One issue of this task is the lack of annotated corpus for evaluation. ToneAnalyzer[15] is a web demo of multiple NLP APIs, including the emotion detection task with 5 Ekman categories: Anger, Disgust, Fear, Joy and Sadness.

Readability Analysis: Readability Analysis task (RA) according to paper [19], the detection of RC consists of determining documents suitable for being read by specific people age ranges. This includes determining reading difficulties or text comprehension. For addressing this task it is necessary to classify documents on the basis of different levels of reading comprehension. Different measures of readability must be selected, the work is based on the study Flesh-Kincaid Grade Level. This study tries to predict the recommended age to understand the text. The tool Readable.io[16] is a web demo for evaluating the readability level of a text using various grade levels, including Flesh-Kincaid Grade Level.

Informality Analysis: Informality Analysis task (IA) tries to detect the degree informality in texts. This task arises due to the necessity of processing in a personalised way non-traditional textual sources existing on the Internet (i.e. blogs, forums, etc.). TENOR [20] provides functionalities aligned to this task.

Age Estimation: Age Estimation task (AE) is part of the Author Profiling area. According to paper [21], AE tries to predict some aspects of authors

[14] http://alchemy-language-demo.mybluemix.net

[15] http://tone-analyzer-demo.mybluemix.net

[16] http://readable.io

like age or gender. This task will contribute to this research the discovery of various authors types depending on age ranges. Age Analyzer[17] is a web API that provides functionalities aligned to this task.

Gender Detection: Gender Detection task (GD) is strongly related to AE task, but in this case, it tries to detect the gender of text's author. Gender Analyzer[18] is appropriated to this task as has been before mentioned.

Irony Detection: Irony Detection task (ID) tries to detect if a literal message has an opposite meaning, without a negation marker. The difficulty resides the absence of face-to-face contact and vocal intonation. In the automatic detection of irony is used sentiment analysis, information extraction or decision making to obtain textual features for recognising irony. Paper [22] presents a research for irony detection in Twitter short documents, using the tasks mentioned above.

Ideology Detection: Ideology Detection task (IDD) tries to detect the ideology orientation expressed in a text content based on a set of opinions or beliefs. Usually, this refers to a set of political beliefs or a set of ideas that characterise a particular culture. Paper [23] presents a research work where political ideology orientation is detected using neural network technologies.

5.1 Automation and Reliability Study

In the present study, the high-reliability degree is defined as technologies that have an evaluation with high scores of performance. Similarly, the high-automation degree is defined as the type of technology easier to implement or use in each case. For example, Web Services, Java or Python libraries, Algorithms, Web application/demo and Desktop tools. Web Services or online APIs present a very-high-automation degree because it is easy to use them in frameworks or meta-tools developments. Java or Python libraries have a high-automation degree, because it is easy to include them in frameworks or meta-tools develops, but sometimes it is needed to proceed with an adaptation stage of the target programming framework. Algorithms present a medium-automation degree because it should be considered the efforts of reproducing them. Web application/demo tools have a low-automation degree because initially, it is difficult to automatically include them, however, an alternative is using web crawling.

Most experimental approaches that make use of Web application/demo tools apply semi-automatic procedures. Desktop tools have a very-low-automation degree because it is very difficult to include them in frameworks or meta-tools develops, most of the time the procedures are performed by hand. Table 1 shows a detailed comparison at respect. The automation of document profiling procedures, such as this study reveals, is supported by the reliability degree presented in Table 1. The performing scores have been taken from the bibliography before cited, being them the best technologies found in the state of the art. The technologies where is set "Not found", probably it is because they represent

[17] http://ageanalyzer.com

[18] http://www.genderanalyzer.com

Table 1. NLP technologies comparisons.

NLP Technology	NLP task	Measure	Score	Type
Dandelion	TC	Not found	Not found	Web demo
DEiXTo	IE	Not found	Not found	Desktop
TextRazor	TR	Not found	Not found	Web demo
MeaningCloud	TR	Not found	Not found	Web demo
SKET	KE	F1	0.7	Java library
Stanford SUTime	TER	F1	0.92	Java library
TIPSem	TER	F1	0.85	Web service
Stanford NER	NER	F1	0.8876	Java library
Summarise	AS	F1	0.57797	Web service
ISR-WN	DD	F1	0.52	Web service
Alchemy Language	LI	Not found	Not found	Web demo
Kiritchenko et al. (2014)	PC	F1	0.855	Algorithm
Stanford Sentiment	PC	Accuracy	0.854	Java library
Zhang et al. (2017)	ED	F1	0.56	Algorithm
Tone Analyzer	ED	Not found	Not found	Web demo
readable.io	RA	Not found	Not found	Web demo
TENOR	IA	Not found	Not found	Algorithm
Age Analyzer	AE	Not found	Not found	Web service
Gender Analyzer	GD	Accuracy	0.95	Web service
Reyes et al. (2013)	ID	F1	0.76	Algorithm
Iyyer et al. (2014)	IDD	Accuracy	0.702	Algorithm

developments related to companies or private research. However, it is interesting to study them for future evaluations. At respect to Web demos, these offer limited NLP services which could be possible resolve through business registration or payment of services. Regarding the type "Algorithm", in some cases, this one is not available in the reference papers. Nevertheless, it is considered because the authors could be contacted in some way.

The study revealed that many NLP technologies are interesting to this aim, however, many of them are difficult to be reused. This depends on the licenses of use, visibility, replicability of the algorithms, etc.

6 Conclusion and Future Work

In this paper is presented the study of useful NLP technologies to automate the process of building document's profiles. Clearly, knowing the difficulties found to reuse the NLP technologies will help us to be more focused on considering those technologies with a high-automation degree instead of a high-reliability degree. As result, in this paper is demonstrated many different NLP technologies can converge all in a unique ecosystem. It can improve retrieval systems and even be able to recommend documents to users. As future work is planned to create a dataset for further supporting the creation and evaluation of document profile. In addition, the study of the results of evaluating different types of document profiles (i.e. social, press, book, etc.) will be included in our research agenda. This study could include the adaptation of some NLP technologies at the proposed work, for instance, a new classification system using the documents' schema.

Acknowledgments. This work is funded by a FPU predoctoral fellowship granted by University of Alicante (UAFPU2015-5999), and has also been partially funded by the University of Alicante, Generalitat Valenciana, Spanish Government, Ministerio de Educación, Cultura y Deporte and Ayudas Fundación BBVA a equipos de investigación científica 2016 through the projects TIN2015-65100-R, TIN2015-65136-C2-2-R, PROMETEOII/2014/001, GRE16-01: "Plataforma inteligente para recuperación, análisis y representación de la información generada por usuarios en Internet" and ASAP.

References

1. Sapkota, U., Bethard, S., y Gómez, M.M., Solorio, T.: Not all character n-grams are created equal: A study in authorship attribution. 2015 Conf. North Am. Chapter Assoc. Comput. Linguist. – Hum. Lang. Technol. (NAACL HLT 2015). 93–102 (2015). doi:10.3115/v1/N15–1010
2. Bobadilla, J., Ortega, F., Hernando, A., Gutiérrez, A.: Recommender systems survey. Knowledge-Based Syst. 46, 109–132 (2013). doi:10.1016/j.knosys.2013.03.012
3. Gulla, J.A., Fidjestøl, A.D., Su, X., Castejon, H.: Implicit User Profiling in News Recommender Systems. Int. Conf. Web Inf. Syst. Technol. 185–192 (2014). doi:10.5220/0004860801850192
4. Usbeck, R.: Combining Linked Data and Statistical Information Retrieval (2014)
5. Cavnar, W.B., Trenkle, J.M., Mi, A.A.: N-Gram–Based Text Categorization. Proc. SDAIR–94, 3rd Annu. Symp. Doc. Anal. Inf. Retr. 161–175 (1994). doi:10.1.1.53.9367
6. Kshirsagar, A.A., Deshkar, P.A.: Review analyzer analysis of product reviews on WEKA classifiers. In: ICIIECS 2015 – 2015 IEEE International Conference on Innovations in Information, Embedded and Communication Systems (2015)
7. Hajeer, S.I., Ismail, R.M., Badr, N.L., Tolba, M.F.: An adaptive information retrieval system for efficient web searching. Commun. Comput. Inf. Sci. 488, 472–482 (2014)
8. Vila, K., Fernández, A., Gómez, J.M., Ferrández, A., Díaz, J.: Noise-tolerance feasibility for restricted-domain Information Retrieval systems. Data Knowl. Eng. 86, 276–294 (2013). doi:10.1016/j.datak.2013.02.002
9. Rossi, R.G., Marcacini, R.M., Oliveira Rezende, S.: Analysis of domain independent statistical keyword extraction methods for incremental clustering. J. Brazilian Soc. Comput. Intell. 12, 17–37 (2014)
10. Finkel, J., Grenager, T., Manning, C.: Incorporating non-local information into information extraction systems by gibbs sampling. 43rd Annu. Meet. 363–370 (2005). doi:10.3115/1219840.1219885
11. Chang, A.X., Manning, C.D.: SUTime: A library for recognizing and normalizing time expressions. LREC. 3735–3740 (2012). doi:10.1017/CBO9781107415324.004
12. Llorens, H., Saquete, E., Navarro, B.: Temporal expression identification based on semantic roles (2009)

13. Alcón, O., Lloret, E.: Estudio de la influencia de incorporar conocimiento léxico-semántico a la técnica de análisis de componentes principales para la generación de resúmenes multilingües (2015)
14. Gutiérrez, Y., Vázquez, S., Montoyo, A.: A semantic framework for textual data enrichment. Expert Syst. Appl. 57, 248–269 (2016). doi:10.1016/j.eswa.2016.03.048
15. Mohammad, S.M.: Sentiment Analysis: Detecting Valence, Emotions, and Other Affectual States from Text. In: Emotion Measurement. pp. 201–237 (2016)
16. Kiritchenko, S., Zhu, X., Mohammad, S.M.: Sentiment analysis of short informal texts. J. Artif. Intell. Res. 50, 723–762 (2014). doi:10.1613/jair.4272
17. Socher, R., Perelygin, A., Wu, J.: Recursive deep models for semantic compositionality over a sentiment treebank. Proc. 1631–1642 (2013). doi:10.1371/journal.pone.0073791
18. Ekman, P.: Basic Emotions. In: Handbook of Cognition and Emotion. pp. 45–60 (2005)
19. Valdivia, M.T.M., Cámara, E.M., Barbu, E., López, L.A.U., Moreda, P., Lloret, E.: Proyecto FIRST (flexible interactive reading support tool): Desarrollo de una herramienta para ayudar a personas con autismo mediante la simplificación de textos. Proces. Leng. Nat. 53, 143–146 (2014)
20. Mosquera, A., Moreda, P.: TENOR: A lexical normalisation tool for spanish web 2.0 texts. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). pp. 535–542 (2012)
21. Rangel, F., Rosso, P.: On the impact of emotions on author profiling. Inf. Process. Manag. 52, 73–92 (2016). doi:10.1016/j.ipm.2015.06.003
22. Reyes, A., Rosso, P., Veale, T.: A multidimensional approach for detecting irony in Twitter. Lang. Resour. Eval. 47, 239–268 (2013). doi:10.1007/s10579–012–9196-x
23. Iyyer, M., Enns, P., Boyd-Graber, J., Resnik, P.: Political Ideology Detection Using Recursive Neural Networks. Acl-2014. 1113–1122 (2014). doi:10.1017/CBO9781107415324.004

ISSN 1870-4069

User Identification in Pinterest Through the Refinement of a Cascade Fusion of Text and Images

Juan Carlos Gomez, Mario-Alberto Ibarra-Manzano, and Dora-Luz Almanza-Ojeda

Universidad de Guanajuato, Departamento de Ingeniería Electrónica, DICIS,
Salamanca, Mexico

{jc.gomez,ibarram,dora.almanza}@ugto.mx

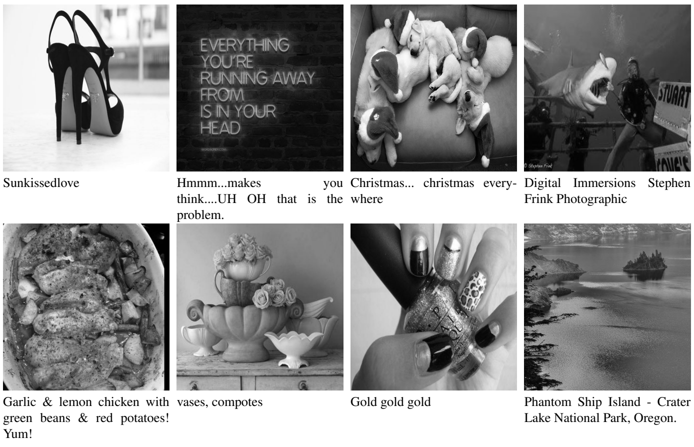
Abstract. User identification in social media is of crucial interest for companies and organizations for purposes of marketing, e-commerce, security and demographics. In this paper, we aim to identify users from Pinterest, a platform where users post *pins*, a combination of an image and a short text. This type of multi-modal content is very common nowadays, since it is a natural way in which users express their interests, emotions and opinions. Thus, the goal is to identify the user that would post a particular pin. For solving the problem, we propose a two-phase classification model. In a first phase, we train independent classifiers from image data, using a deep learning representation, and from text data, using a bag-of-words representation. During testing we apply a cascade fusion of the classifiers. In a second phase, we refine the output of the cascade for each test pin by selecting the top most likely users for the test pin and re-weighting their corresponding output in the cascade by their similarity with the test pin. Our experiments show that the problem is very hard because several reasons with the data distribution, but they also show promising results.

Keywords. Social media, user identification, pinterest, deep learning, information fusion.

1 Introduction

In the Big Data era, a large amount of information is created and transmitted around the world through the Internet [3]. Much of the traffic occurs on social media and similar platforms, where users create and share multimedia content such as news, reports, videos, emotions, music, opinions, etc. The content generated by users in social media has some particularities [26,34]: is plentiful, is constantly generated, is dynamic (shared and distributed by the users in real time), is representative of users or groups of users, and is multi-modal [27], meaning it is composed by a mixture of text, images, videos, audio and links that connect users and websites (friends, followers, shares, reactions, related websites, etc.). All the information that users generate in social media is a tool that can help to draw specific user profiles [21,7], and represents a digital footprint that identify how the persons use the social media, indicating their tastes, likes, preferences, personalities, sentiments, types of friendships or connections, etc.

Currently, there is a great interest from companies and organizations to analyze user generated content in social media, with the purpose of obtaining useful indicators for



Sunkissedlove

Hmmm...makes you think....UH OH that is the problem.

Christmas... christmas everywhere

Digital Immersions Stephen Frink Photographic

Garlic & lemon chicken with green beans & red potatoes! Yum!

vases, compotes

Gold gold gold

Phantom Ship Island - Crater Lake National Park, Oregon.

Fig. 1. Examples of pins from Pinterest. Each pin is formed by an image and a short comment about the image.

making decision on several areas [6,30,9], such as security, politics, social, educational, commercial and marketing. In this direction, user identification serves to several purposes. For companies and organizations, it helps to group persons by similar interests and behavioral patterns, recognize lead users, influencers, potential customers, political sympathizers and even to detect trolls, intruders, terrorists or persons that are a threat for public security [5]; considering that in social media users can assume multiple or false identities. Additionally, users can benefit by receiving personalized content that is adequate for their needs of entertainment, shopping, security, health and education.

Social media posts present several characteristics that are challenging for user identification. First, they are multi-modal, since this is a natural way for persons to express an opinion, emotion or to share content. This type of content suffers of a semantic gap between modalities, since what is expressed in text could not be representative of what is shown in an image. Second, users in general do not follow grammar and spelling rules, which makes difficult to use high-level language attributes such as syntax and semantics [21]. Third, text varies largely in length among posts. Fourth, text and images are highly heterogenous, including a large diversity of topics. Fifth, there are many users, and their content could overlap.

In this work, we present a two-phase model for user identification in Pinterest, which is one of the most popular social networks in the world with over 150 million of monthly active users. In Pinterest, users post *pins*, a combination of an image and a short text about the image. Figure 1 presents some example of pins, showing the diversity of content that we can find in this social site.

We define our task as identifying the specific user that would have post a pin (combination of image and text). We pose the problem as a single-label multi-class classification task, where the classes correspond to specific users and a pin could belong to only one user. For solving the problem, we propose a two-phase classification model. In a first phase, we train independent classifiers from image data, using a deep learning representation, and from text data, using a low-level bag-of-words representation. During testing, we apply a cascade fusion of the classifiers, where the probabilities outputs of both classifiers are multiplied sequentially. The cascade fusion classifier is aimed to combine the two data modalities for reducing the semantic gap and exploiting better the whole content. In a second phase, we refine the output of the cascade fusion for each test pin. The refinement is done per test pin, selecting the top most likely users for such pin, and then adjusting their corresponding probability scores by measuring the maximum similarity of all their training pins with the test pin. Our experiments confirm that the task is highly complex, due to the reasons explained above, but also show promising results.

The rest of the paper is organized as follows. Section 2 briefly reviews the related works. Section 3 presents our methodology, including the data representations and the two-phase classification model. Section 4 describes the setup used for experimentation, including the dataset and the evaluation metrics. Section 5 shows the results. Finally, Sect. 6 concludes the paper with an overall discussion and future research directions.

2 Related Work

User identification has been conducted using different representations for the user generated data, but most of the works are based exclusively on text data [21,28,31], image data [13,32], link data [35,16] and metadata [29]. In this work, we aim to combine multi-modal data from text and images trying to exploit most of the content for a better user identification.

There exist several approaches for user identification [1]. In [7,2,20] clustering techniques such as K-means, DBSCAN and normalized mutual information are used as a first approach for grouping users with similar patterns, and then identified particularities of each one. Similarly, classification methods such as K-Nearest Neighbors [21], Gradient Boosted Decision Trees [23] and Support Vector Machines [17] have been used for identifying user based on their profiles. In addition, some works follow an information retrieval approach and estimate distance metrics for ranking possible users for a given content [22,33]. In this work, we employ a single-label multi-class model approach, where each class correspond to a specific user, and the content could be assigned exclusively to one user.

Image and text data can be represented in different manners. In text, while high-level [24] and deep learning [18] text features can reflect users' writing styles and syntactic and semantic relations [19], lower-level text features such as words frequencies and n-grams could produce similar results with a lower cost. For example, in [4] the authors found that a bag-of-words representation outperforms the deep learning representation of word2vec [18] when inferring users' interests in Pinterest. In images, the state of the art is to use deep learning features, mainly using Convolutional Neural Networks (CNN), that have achieved good performance in many image classification and object detection tasks [15,25], as well as in image description generation [14] and automatic comments generation [12]. In this work, we employ a bag-of-words representation of text and a deep learning representation of images, both for its good results working with similar data.

Our work is particularly close related to [4], where the authors present a model that linearly combines the output of two independent classifiers trained over the modalities of text and images for inferring user interests in Pinterest. We use their model as a baseline for comparison with ours, considering that we are trying to identify specific users, rather than their interests. User identification is a more challenging task because of the number of possible classes involved. Additionally, we use as baselines models that are built using exclusively text or image data.

3 Methodology

In Pinterest, users post *pins* which are organized in boards. A pin is an ¡image,text¿ pair, and a board groups several pins that represent particular users' interests. Here we discard the board information, and put all the pins from a user in a single collection. Be u a user and $\mathbf{P}_u = (\mathbf{g}_i^u, \mathbf{x}_i^u)$ his collection of pins, where $\mathbf{g}$ refers to the image and $\mathbf{x}$ to the text. There are $i = 1, \ldots, n_u$ pins per user, m total users and $n = \sum_i = 1^m (n_{u_i})$ total pins. The task consists in identifying the user $\widehat{u}$ that would have generated a pin $p_t = (\mathbf{g}_t, \mathbf{x}_t)$

We stack all the pins of all the users in a single dataset, and we split it in a training, a validation and a test set, keeping the same proportions for each user. With text, we first cleaned each pin by removing special symbols (e.g. hash tags, asterisks, etc.), stop words, URLs, one-letter words and long words (> 30 characters). Using exclusively the training set we extract a dictionary removing words appearing in only one pin. Afterwards, we use a tf-idf weighting schema to build document-term matrices $\mathbf{X}_{tr}$, $\mathbf{X}_v$ and $\mathbf{X}_t$ for each part of the data. We preprocessed each image using a Convolutional Neural Network to obtain the corresponding row vector $\mathbf{g}^u$ of features for user u. We used for that the DeCAF library [8], considering the activation values of the 4,096 neurons in the 7-th layer as the image features. The DeCAF model used for the transformation of images, was pre-trained with the Imagenet dataset [15]. We join all the image vectors corresponding to the training, validation and test sets in matrices $\mathbf{G}_t r$, $\mathbf{G}_v$ and $\mathbf{G}_t$ that are row paired with matrices $\mathbf{X}$ and sorted by user.

3.1 Classification Model

Our classification model consists of two phases. In the first phase, we train two independent classifiers using separately the pins' text data $\mathcal{F}_{\mathbf{x}}$ and image data $\mathcal{F}_{\mathbf{g}}$ considering we have m different users representing the classes. In this work, we employ logistic regression as the individual classifier, since this model can be trained naturally for a multi-class problem, and directly outputs a probability for each user. The models are first built using the training sets $\mathbf{X}_{tr}$,$\mathbf{G}_{tr}$ and use the validation sets $\mathbf{X}_v$,$\mathbf{G}_v$ for independently optimize the regularization parameter C, considering the values [0.1,1,10,100]. After optimization, we merge the training and validation sets in two single sets and train independent models with the whole data using the optimal C.

During testing, for each test pin $p_t = (\mathbf{g}_t, \mathbf{x}_t)$ we classify independently its text part as $\mathcal{F}_{\mathbf{x}}(\mathbf{x}_t)$ and its image part as $\mathcal{F}_{\mathbf{g}}(\mathbf{g}_t)$, producing the corresponding two vectors of probabilities of belonging to each user: $\mathbf{r}_x = [r_{u_1}^x, r_{u_2}^x, \ldots, r_{u_m}^x]$ and $\mathbf{r}_g = [r_{u_1}^g, r_{u_2}^g, \ldots, r_{u_m}^g]$. Afterwards, we combine the two outputs in a cascade fusion as:

$$\mathbf{r} = \mathbf{r}_x \cdot \mathbf{r}_g. \tag{1}$$

This equation outputs a fusion probability vector $\mathbf{r} = [r_{u_1}, r_{u_2}, \ldots, r_{u_m}]$ that merges together the two data modalities. We can select from this vector the user $\widehat{u}$ for the test pin p_t as the one with the highest probability.

In a second phase, for the test pin p_t we extract the top 10 most likely users $top = [\tilde{u}_1, \tilde{u}_2, \ldots, \tilde{u}_10]$ and their fusion probabilities $\mathbf{r}_{top} = [r_{\tilde{u}_1}, r_{\tilde{u}_2}, \ldots, r_{\tilde{u}_10}]$. Using the training pins of each top user $\mathbf{P}_{\tilde{u}_i}$ we compute the cosine similarity of all their pins with the test pint p_t and find the maximum similarity $l_{\tilde{u}_i}$, and re-weight its corresponding probability in vector $\mathbf{r}_{top}$ as $r_{\tilde{u}_i} = r_{\tilde{u}_i} * l_{\tilde{u}_i}$. Finally, from vector $\mathbf{r}_{top}$ we select user $\widehat{u}$ for the test pin p_t as the one with the highest value.

4 Setup

In this work, we use a dataset of 70,200 pins belonging to 117 users, that were randomly collected by directly crawling the Pinterest website[1]. Thus, there are 117 different classes. We selected at random three boards per user, saving 200 pins per board. We discarded the board information and merged all the pins for a total of 600 pins per user. For the experiments, we split the dataset selecting at random 400 pins per user for training, 100 pins per user for validation and 100 pins per user for testing.

All the pins contain and image and a text. The comments are in English and are of a variable length from one (12.33% of the pins) to a maximum of 552 words. The dictionary to build the document-temr matrix was extracted from the training set during validation and from training and validation sets when building the final model. The final dictionary was composed of 17,145 words. In Table 1, we show the five top common words, with their pin frequency, used by five random users. We observe there the diversity of topics the users talk about. In Figure 2, we show statistics about the minimum, maximum, median and average number of words in pins per user (using a logarithmic scale). The distribution of words is generally regular, with some users having larger comments, considering that the minimum and the median are similar. The statistics for the number of words in pins in the whole dataset are a minimum of 1, a maximum of 552, a median of 4 and average of 8.5.

Table 1. Most frequent words for a sample of 5 users; indicating the number of pins for that user on which such word appears.

User 1	User 2	User 3	User 4	User 5
logo(77)	make(45)	love(28)	crochet(107)	diamond(155)
design(64)	diy(34)	vintage(20)	pattern(56)	ring(144)
via(51)	cream (32)	one(14)	free(48)	necklace(127)
infographic(46)	chicken(30)	black(12)	com(33)	gold(82)
designspiration(46)	cake(30)	elizabeth(10)	art(30)	sapphire(43)

[1] Dataset is available at: `https://data.mendeley.com/datasets/fs4k2zc5j5/2`

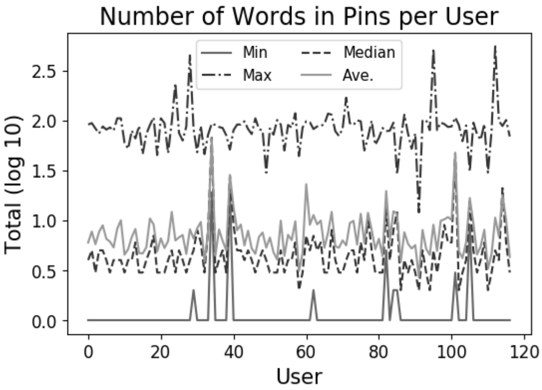


Fig. 2. Words statistics in pins over users.

The images in the dataset also vary in size, but they are all in JPG format. Only 4.2% of images are shared by 2 or more users, which shows the great diversity of content in the pins, including products (e.g. clothes and jewelry), interests (e.g. food and decoration), photographs (e.g. animals and landscapes) and more abstract content (e.g. paints and designs).

Text and images even if coupled, represent different modalities of data, and between them exist a semantic gap, since a comment about an image could not be expressing the same meaning as the visual depiction. That is, a text comment could be applied to several images, and an image could be described/commented in many ways. In Figure 3 we show the whole of text and image data projected independently over the two principal components (PC) after transforming the data using principal component analysis (PCA). In the figure, we observe how the text and image data are distributed differently along the PCs. For text data, the general explained variance using 10 PCs is 0.043, while for image data is 0.321. Images in our dataset are more compact in their features than text, and could be better compressed using PCA. That means that their features are more homogenous, even if its content is properly highly heterogenous. This could serve to reduce the feature space and simplify some computations, but also it does not help for separating data of different users. On the other hand, text data is more heterogenous in its features, like its content, and could not be well compressed with PCA, but it serves to better distinguish the data from different users up to a certain level.

4.1 Baselines and Evaluation

As first baselines, we use the two logistic regression models that are built using separately the pins' text data $\mathcal{F}_{\mathbf{x}}$ and image data $\mathcal{F}_{\mathbf{g}}$. Such models are the base for the cascade fusion model and are optimized for the regularization parameter C, using the validation set and considering values of $C = [0.1, 1,]$. The other set of baseline models are taken from [4]. In that work, the authors built a model that when testing

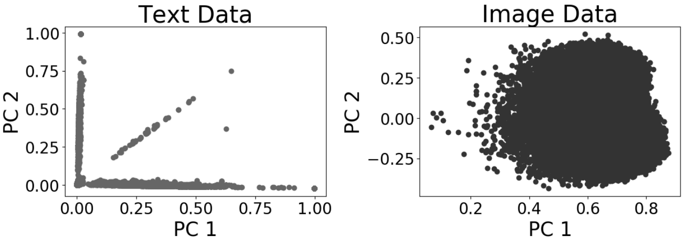


Fig. 3. Distributions of text and image data over two principal components.

for a pin_t, it takes the output of classifiers $\mathcal{F}_{\mathbf{x}}$ and $\mathcal{F}_{\mathbf{g}}$, and using a late fusion schema, weight the contribution of each one depending on a parameter λ, as follow $pred_{p_t} = \lambda\mathcal{F}_g(\mathbf{g}_t) + (1-\lambda)\mathcal{F}_x(\mathbf{x}_t)$. We consider three values for $\lambda = [0.3, 0.5, 0.7]$, as in the mentioned work.

We compare the performance of all the methods using the standard classification measures: accuracy and F1. These are defined as $accuracy = \frac{(tp+tn)}{(tp+fp+fn+tn)}$ and $F1 = 2\frac{precision \cdot recall}{precision+recall}$, where tp are the true positives, tn the true negatives, fp the false positives and fn the false negatives. Accuracy measures the proportion of corrected identified users, while the F1 measure represents the harmonic mean of precision and recall, that are in turn defined as: $precision = \frac{tp}{tp+fp}$, $recall = \frac{tp}{tp+fn}$. We compute the macro averages for F1, since in our case the classification is single-label, and the micro averages of F1 are equal to the accuracy [11].

We ran all our experiments using a Windows PC with a 2.5 GHz Core i5 processor and 8 GB in RAM. We implemented all our methods in Python[2], using the scikit-learn[3] and NumPy[4] libraries.

5 Results

Table 2 shows the summary of results for all the models. We can see than in general, the performance of the models is low, with less than 40% for accuracy and F1, making clear that the problem is hard to solve. We observe that using only images for user identification produces the lowest results. As mentioned before, image data is more homogenous on its features, but not on its content, and could have problems to separate the data of different users. We believe this is associated with the transformation done with the DeCAF library, which is trained using a generic dataset of images (Imagenet). We guess that a fine tuning of the DeCAF model with images from Pinterest could help to improve this performance. When using only text, the results are better, as expected from the same argument as before, text reflects better the heterogeneity of the data and can discriminate better the data of different users. Our cascade fusion (CF) model performs better than all the baselines, with near 4% of improvement for accuracy and

[2] Code is available at: `https://github.com/jcgcarranza/2017rcs_code`

[3] `http://scikit-learn.org`

[4] `http://www.numpy.org/`

F1 regarding text, and 1% for F1 regarding the $\lambda = 0.3$ model. The cascade fusion model reaches a recall@10 of 75%, that would be the maximum accuracy performance expected if we ordered correctly the 10 most likely users per test pin. When we use refinement over the cascade fusion model (CF+Ref) the results improve in about 1% more for accuracy and F1. It is thus clear, that the cascade fusion takes better advantage of the information coming from both modalities.

Table 2. Results for the different models with the different metrics.

Model	**Accuracy**	**F1**
Text only	33.75	33.21
Images only	21.62	19.72
[4] $\lambda = 0.3$	37.18	35.01
[4] $\lambda = 0.5$	34.68	32.11
[4] $\lambda = 0.7$	30.73	28.10
CF	37.34	36.16
CF+Ref.	**38.30**	**37.46**

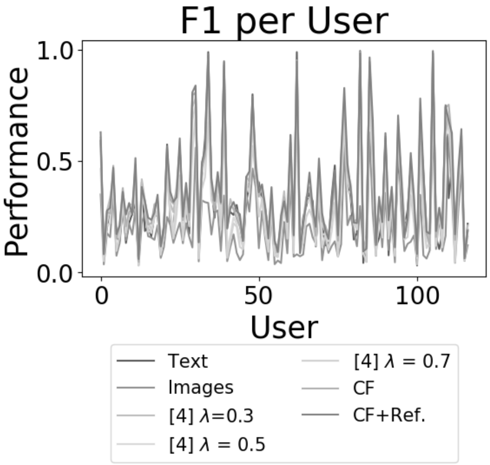


Fig. 4. Performance in F1 for each user.

To better understand the low performance in the task, in Figure 4 we show the performance of the different models per user for F1. All the models present a similar behavior along the different users, with the cascade fusion model and the refined model showing plots a little above than the others. We observe in this plot that some users are more easily identified than others. There are seven users for which the performance is less than 10%, and there are 6 for which the performance is more than 90%. In Figure 5 we plot a histogram of the F1 performance for the CF+Ref model. We can see that for most of the cases user identification is difficult, with general performances below the 50%.

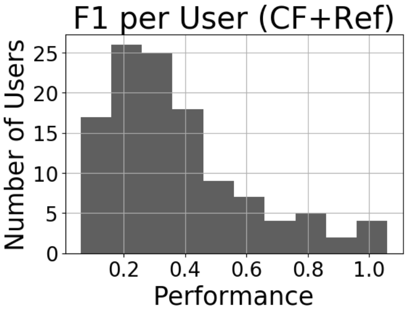


Fig. 5. Histogram of the performance in F1 for the CF+Ref model.

Table 3. Users with the lowest and the highest F1 performance for the CF+Ref model, paired with the median number of words in their pins.

Lowest Performance			Highest Performance		
User	**Median**	**F1**	**User**	**Median**	**F1**
13	4	0.059	106	16	0.995
116	6	0.062	83	13.5	0.985
2	5	0.081	34	68	0.980
101	6	0.083	86	12	0.966
64	5	0.086	40	28	0.949

In Table 3 we show the top five users for which the performance is the lowest and the top five for which it is the highest, showing the F1 metric value and the median length of their pins. Users for which the performance is low, have pins with medians between 4 and 6, while users for which the performance is high have pins with medians between 13.5 and 68. Indeed, considering all the users there is a weak Pearson correlation (0.46) between the performance and the median length of the comments in pins. This means that in general, users that post longer comments provide with more information to better identify them. The general median for all the users is 4, which partially would explain the low performance for most of the users, meaning that users would tend to post very short comments in Pinterest. A second part to explain the low performance comes from the images. As mentioned before, the deep learning features are good for representing the data, but not for discriminate the data of different users. Moreover, in this case for images it is more complicated to analyze the features in detail, since the used image features are convolutions of other more general features, and the semantic meaning is lost in the process. Another part playing in the low performance is the number of users, there are 117 in total, and from Figure 3 we observe that even if the content shows heterogeneity, the distributions of text and image data could be similar among different users.

6 Conclusions and Research Directions

In this work, we have presented a two-phase classification model for user identification in Pinterest, a very popular social media site where users post pins (a mixture of an image and a companion text comment). Our task consisted on identifying the specific user that would have generated a pin. We treated the problem as a single-label multi-class classification task, where the classes correspond to specific users and a pin could belong to only one user. Our classification model in a first phase it trains independent classifiers from image data, using a deep learning representation, obtained from a convolutional neural network mode, and from text data, using a low-level bag-of-words representation weighted using a tf-idf schema. During testing, we apply a cascade fusion of the classifiers, where the probabilities outputs of both classifiers are multiplied sequentially. The cascade fusion model tries to combine the two data modalities for reducing the semantic gap and exploiting better the whole content. In a second phase, we refine the output of the cascade fusion for each test pin. The refinement is done by selecting the top most likely users for the test pin, and then adjusting their corresponding probability scores by measuring the maximum similarity of all their training pins with the test pin.

We tried our model with a dataset of 70,200 pins from 117 users, and compare it with some baselines from the literature. The results showed that the task is very hard, since all the models reached less than 40% of performance for accuracy and macro F1. When analyzing individual performance per user, we observe that only a few of them (6) are identified with more than 90% of accuracy, while most of them are identified with less than 50% of accuracy. In the lowest extreme, there are 7 users for which the performance is less than 10%. In general, we observe a weak Pearson correlation between the length of the user pins (as measured by the median length) and the associated performance. Additionally, image features obtained by a convolutional neural network are good for representing the data, but not for discriminate the data of different users. Text features reflex better the heterogeneity of the data, but also present similar distributions among users. Finally, the number of possible classes is also a challenge, since there is a bigger chance than the distribution of images and text overlap between users.

Future research directions include the use of a DeCAF model fine-tuned with data collected from Pinterest, for computing the image features. Additionally, we guess that it is necessary to transform both modalities of data to other feature spaces, where the data would be better separate [10]. This could be done for each modality independently or both at the same time. Finally, other late fusion models could be explored to combine the outputs of independent classifiers, especially considering that the recall@10 of the cascade fusion model is 75%, meaning that between the 10 most likely users selected per test pin, there is a 75% of chance of finding the correct one. Thus, a better model to re-order the selected users could help to improve the performance.

Acknowledgments. This research was supported in part by PRODEP-SEP project folio UGTO-PTC-595, oficio 511-6/17-8074, by CONACYT-INEGI project Fondo Sectorial Conacyt-Inegi No. 290910, and by Universidad de Guanajuato.

References

1. Barforoush, A.A., Shirazi, H., Emami, H.: A new classification framework to evaluate the entity profiling on the web: Past, present and future. ACM Computing Surveys (CSUR) 50(3), 39 (2017)
2. Cerquitelli, T., Servetti, A., Masala, E.: Discovering users with similar internet access performance through cluster analysis. Expert Systems with Applications 64, 536–548 (2016)
3. Chen, M., Mao, S., Liu, Y.: Big data: A survey. Mobile Networks and Applications 19(2), 171–209 (2014)
4. Cinar, Y.G., Zoghbi, S., Moens, M.F.: Inferring user interests on social media from text and images. In: Data Mining Workshop (ICDMW), 2015 IEEE International Conference on. pp. 1342–1347. IEEE (2015)
5. Cohen, K., Johansson, F., Kaati, L., Mork, J.C.: Detecting linguistic markers for radical violence in social media. Terrorism and Political Violence 26(1), 246–256 (2014)
6. Criado, J.I., Sandoval-Almazan, R., Gil-Garcia, J.R.: Government innovation through social media. Government Information Quarterly 30(4), 319–326 (2013)
7. van Dam, J.W., van de Velden, M.: Online profiling and clustering of facebook users. Decision Support Systems 70, 60–72 (2015)
8. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: Decaf: A deep convolutional activation feature for generic visual recognition. In: International conference on machine learning. pp. 647–655 (2014)
9. Fan, W., Gordon, M.D.: The power of social media analytics. Communications of the ACM 57(6), 74–81 (2014)
10. Gomez, J.C., Boiy, E., Moens, M.F.: Highly discriminative statistical features for email classification. Knowledge and information systems 31(1), 23–53 (2012)
11. Gomez, J.C., Moens, M.F.: Minimizer of the reconstruction error for multi-class document categorization. Expert Systems with Applications 41(3), 861–868 (2014)
12. Gomez, J.C., Tommasi, T., Zoghbi, S., Moens, M.F.: What would they say? predicting user's comments in pinterest. IEEE Latin America Transactions 14(4), 2013–2019 (2016)
13. Hum, N.J., Chamberlin, P.E., Hambright, B.L., Portwood, A.C., Schat, A.C., Bevan, J.L.: A picture is worth a thousand words: A content analysis of facebook profile photographs. Computers in Human Behavior 27(5), 1828–1833 (2011)
14. Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3128–3137 (2015)
15. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012)
16. Liu, S., Wang, S., Zhu, F., Zhang, J., Krishnan, R.: Hydra: Large-scale social identity linkage via heterogeneous behavior modeling. In: Proceedings of the 2014 ACM SIGMOD international conference on Management of data. pp. 51–62. ACM (2014)
17. Malhotra, A., Totti, L., Meira Jr, W., Kumaraguru, P., Almeida, V.: Studying user footprints in different online social networks. In: Advances in Social Networks Analysis and Mining (ASONAM), 2012 IEEE/ACM International Conference on. pp. 1065–1070. IEEE (2012)

18. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
19. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems. pp. 3111–3119 (2013)
20. Mislove, A., Viswanath, B., Gummadi, K.P., Druschel, P.: You are who you know: inferring user profiles in online social networks. In: Proceedings of the third ACM international conference on Web search and data mining. pp. 251–260. ACM (2010)
21. Peng, J., Choo, K.K.R., Ashman, H.: Bit-level n-gram based forensic authorship analysis on social media: Identifying individuals from linguistic profiles. Journal of Network and Computer Applications 70, 171–182 (2016)
22. Peng, J., Detchon, S., Choo, K.K.R., Ashman, H.: Astroturfing detection in social media: a binary n-gram–based approach. Concurrency and Computation: Practice and Experience 29(17) (2017)
23. Pennacchiotti, M., Popescu, A.M.: A machine learning approach to twitter user classification. Icwsm 11(1), 281–288 (2011)
24. Ratkiewicz, J., Conover, M., Meiss, M., Gonçalves, B., Patil, S., Flammini, A., Menczer, F.: Detecting and tracking the spread of astroturf memes in microblog streams. arXiv preprint arXiv:1011.3768 (2010)
25. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International Journal of Computer Vision 115(3), 211–252 (2015)
26. Russell, M.A.: Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More. O'Reilly Media, Inc. (2013)
27. Scott, J.: Social network analysis. Sage (2017)
28. Shen, X., Tan, B., Zhai, C.: Implicit user modeling for personalized search. In: Proceedings of the 14th ACM international conference on Information and knowledge management. pp. 824–831. ACM (2005)
29. Sloan, L., Morgan, J., Burnap, P., Williams, M.: Who tweets? deriving the demographic characteristics of age, occupation and social class from twitter user meta-data. PloS one 10(3), e0115545 (2015)
30. Tan, W., Blake, M.B., Saleh, I., Dustdar, S.: Social-network-sourced big data analytics. IEEE Internet Computing 17(5), 62–69 (2013)
31. Tuarob, S., Tucker, C.S.: Automated discovery of lead users and latent product features by mining large scale social media networks. Journal of Mechanical Design 137(7), 071402 (2015)
32. Wu, Y.C.J., Chang, W.H., Yuan, C.H.: Do facebook profile pictures reflect user's personality? Computers in Human Behavior 51, 880–889 (2015)
33. Yang, Y.C.: Web user behavioral profiling for user identification. Decision Support Systems 49(3), 261–271 (2010)
34. Zafarani, R., Abbasi, M.A., Liu, H.: Social media mining: an introduction. Cambridge University Press (2014)
35. Zafarani, R., Liu, H.: Connecting users across social media sites: a behavioral-modeling approach. In: Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 41–49. ACM (2013)

ISSN 1870-4069

Assessment of the Emotional State in Domestic Dogs Using a Bi-dimensional Model of Emotions and a Machine Learning Approach for the Analysis of its Vocalizations

Humberto Pérez-Espinosa[1,2], Verónica Reyes-Meza[3],
María de Lourdes Arteaga-Castañeda[3], Ismael Espinosa-Curiel[2],
Amando Bautista[3], Juan Martínez-Miranda[1,2]

[1] Mexican National Research Council (CONACyT), Mexico

[2] CICESE-UT, Tepic, Nayarit, Mexico

[3] Universidad Autónoma de Tlaxcala,
Centro Tlaxcala de Biología de la Conducta, Mexico

hperez@cicese.mx, vrmeza@gmail.com

Abstract. Artificial intelligence techniques have been used to classify automatically animal vocalizations. The researchers have found distinct acoustic patterns from the analysis of the relationship between regular and irregular components of the signal. In this work, we analyzed the feasibility of training models for bark classification according to the emotional state of the dog using a model designed to represent human emotions. For this, we induced positive and negative emotions in 10 Schnauzer and 10 Chihuahua dogs reacting to the presence of a stranger and playing with its owner. We recorded and analyzed almost six thousand audio samples of dog vocalizations. Human evaluators assessed the emotional status of dogs. We also found a strong correlation between the evaluators' assessments and the automatic estimation of valence and arousal. Our study underlines the validity of a new technique to classify dog's emotional states using an automatized acoustic analysis of its vocalizations that could be useful to model the relationship between sound properties and emotional content.

Keywords. Automatic barks classification, dog vocalizations analysis, dog's barks classification.

1 Introduction

The emotional perception is a critical trait during the life of social mammals; this trait is fundamental for eliciting approach or provoking rejection towards stimuli [3]. In humans, as in other mammals, emotional states have been studied using behavioral components. However, it is possible to measure emotional valence (positive or negative) from vocalizations [14, 5, 8, 4]. Studies in human voice had

pp. 53–65; rec. 2017-09-05; acc. 2017-10-20

found that it is possible to infer emotions from their affective prosody [20], even some models of speech production applied to animal vocal communication, give information about negative or positive characteristics of vocal production and their acoustic structure. For humans, one of the most common animal vocalization is dog barks which have acoustic features such as frequency, amplitude, pitch, and rhythm. Using these parameters we can recognize barks associated with particular contexts (*e.g.*, the presence of a stranger, stay alone, play with the owner) [10, 28, 29, 22]. In a research work conducted by [23], human listeners rated the emotional state of dogs from bark samples using a 5-item emotional scale (aggressiveness, fearfulness, despair, playfulness, and happiness). In most of the cases, there was a correspondence between the categories assigned and the emotional state of the dog. In a similar study [21], children associated barks to emotional facial expressions. The fact that humans can judge the emotional state of dogs based on the acoustic characteristics of its bark is not surprising since we use similar acoustic characteristics to express anger, fear, happiness, and other emotions [1]. The analysis of the dogs' vocalizations can be automated through computer systems to improve the care for pets and even use them for safety-oriented applications or assistance. Some researchers have applied artificial intelligence techniques for automatic classification of barks. They have trained pattern recognition models to differentiate barks previously recorded in different contexts. In a study conducted with over 6,000 bark samples of Mudi dogs, a Bayesian classifier was trained to solve two tasks, recognition of dogs from its barks and barks classification according to their context [17]. They built a set of acoustic descriptors using an evolutionary algorithm and feature selection techniques. The algorithm classified barks according to the context (recording scenario) with an accuracy of 43%. In the same study, the authors trained a model for identifying individual dogs. In this case, they obtained an accuracy of 52%. The contexts with the best classification accuracy were *the presence of a strange* and *dogs fight*. Using the same database, [16] made a comparison of four supervised machine learning methods for the classification tasks: sex, age, context and identification of the dog. In the case of context classification (alone, ball, fight, food, play, stranger, walk), they tested two learning schemes, a single model for all dogs and a model for every dog. Through these experiments, the researchers have proven the feasibility of training models to recognize and classify different aspects of barks. However, the obtained accuracy seemed not enough to create robust applications. It is necessary to further research in data generation, training and artificial intelligence techniques to achieve better results. In the present study, we describe an experimental procedure to generate a corpus of dog vocalizations and to test if it is possible to develop automatic classifiers of dog vocalizations using a bi-dimensional model designed to classify human emotions [19].

2 Materials and Methods

The first step was to generate a database of dog vocalizations. We recorded audio and video from 23 dogs of two breeds: 11 Schnauzer (seven males and four females), and 12 Chihuahua (five males and seven females). The ages ranged from 5 to 84 months.

2.1 Experimental Procedure

Based on related works [21, 9], we designed a protocol for capturing negative emotions such as fear or anger and positive emotions such as happiness or joy. After several pilot tests, we chose the stimuli related to negative (aggression to unfamiliar people) and positive emotions (playfulness). We recorded the dogs in their habitual environment. Before recording, the experimenter contacted the owners and explained to them the objective of the recording session and the procedure. When the owners allowed the experimenter to leave the camera recording at their homes, we filmed the dogs for a couple of hours with the objective of obtaining the barks spontaneously when a stranger knocked at the door, or when the dog played. Other dogs were recorded only during the induction of stimuli (15-25 min).

Aggression to unfamiliar people We divided this procedure into two parts:

Aggr1: The experimenter activates the recording camera before knocking the door. Usually, the dog starts barking when it realizes the presence of a stranger at the door. After the door is open, the experimenter stimulates aggressive barks by doing some threatening movements in front of the dog while he continues recording until the dog stop barking.

Aggr2: To record the second stimulus, the experimenter moves away from the owner's house for ten minutes and then returns. The experimenter pretends to be a thief, hits the door and tries to force it to open. In this case, the barking is more intense. The recording stopped until the dog becomes quiet.

Playfulness We recorded this context after dogs became familiar with the experimenter. Such as in previous studies [26], the experimenter asked the owner to play with the dog using a toy or its favorite object. Then, the experimenter records the interaction between the owner and the dog. However, not all dogs responded to this stimulus.

2.2 Post-processing of Recordings

The post-processing phase included three steps:

1) Manual material selection: the experimenters selected the parts of the audio recordings to include in the study according to their quality. They discarded the parts with excessive ambient noise or low sound volume. We used extracted the audio from videos. **2) Automatic segmentation:** we implemented a method in Matlab R2014b to extract audio segments from the original recordings. This method takes as input the energy and the spectral centroid thresholds to detect audio activity, generating segments of variable lengths that range from very short individual bark sounds (0.4 seconds) to longer segments (4 seconds) that contain a group of 10 individual bark sounds emitted very quickly without any pause. **3) Manual classification:** the experimenters classified each generated audio segment and stored in a particular directory according to each of the nine classes listed in Table 1.

The experimenters who carried out this classification process knew beforehand what stimuli caused each vocalization. Based on this knowledge, they performed the manual classification process. The only category where they had to use their subjective assessment was to discriminate between aggressive and very aggressive barks. The rest of the categorization tasks were performed objectively. At the end of this process, we obtained 10:43:27 hours of audio-visual material. The corpus consists of 5,805 audio segments: 167 whines, 61 growls and 3,209 barks (1,822 aggressive, 1,036 very aggressive, 192 in the context of play and 159 that were not induced by the protocol).

Table 1 shows the classes of sounds from the analyzed segments:

Table 1. Classes of sounds.

Vocalization	***Desciption***
Positive-growl	Growl generated by positive stimuli.
Negative-growl	Growl generated by negative stimuli.
Positive-whine	Whine generated by positive stimuli.
Negative-whine	Whine generated by negative stimuli
Aggressive-bark	Bark occurred due to the presence of a stranger.
Very-Aggressive-bark	Bark occurred due to the presence of a stranger and imminent threat.
Play-bark	Bark occurred while the dog is playing.
Other-bark	Bark that was not produced by some of the planned stimuli.
Other-sound	All other sounds from the background (TV, human voices, bird sounds, etc).

2.3 Human Assessment of Recordings

To assess the emotional content in the vocalizations, we used a continuous scale of emotional vocalizations inspired in the Self-Assessment Manikin (SAM) scales [2]. The SAM is based on a three-dimensional model of human emotions. The dimensions are the arousal, valence, and dominance. Some authors named these dimensions as emotion primitives [27, 11]. They argue that emotions are built on a small number of basic emotion descriptors that are not themselves emotions, but all emotions correspond to combinations of these primitives [6]. We worked with two of these primitives, arousal, and valence which are the most commonly used. We decided to use only two primitives to simplify the model. Arousal describes the perceived vigilance as a physiological and psychological condition.

For example, boredom is supposed to have low arousal, and ecstasy is supposed to have high arousal. Valence describes the positive or negative feeling caused by a situation, an object or an event (*e.g.*, anger is supposed to have a negative valence, and joy is expected to have positive valence). We are interested in finding out these emotion primitives in dog vocalizations.

Participant judges 101 human judges participated in listening sessions. 77 female and 24 male, from 17 to 55 years old, mean age of 21. All the participants had lived with dogs from one to 30 years (mean 11 years). The vocalizations samples were played back on a computer by an assistant. At the beginning of the evaluation session, we distributed the answer sheets to each judge. The arousal and valence scale was explained to the judges, clarifying its use with examples.

Playback stimuli We selected the ten dogs with more samples from each of the two breeds. From the original recordings, we randomly extracted four 10-seconds-long segments for each dog. We excluded samples with environmental noise that could interfere with the assessment of emotion primitives (e.g., knock the door, human voice, and sounds from TV). In total, we used 40 segments of each breed. 50 human judges listened only to the Chihuahua samples, and 51 listened only to the Schnauzer samples (in the same order). After listening to each audio segment, the judges filled a sheet that shows the dog emotional scale. The judges had to assess both valence and arousal for each playback. We performed the playback test in small groups (3 - 6 people). It is important to mention that the judges do not know what stimulus provoked the barking they are hearing.

2.4 Acoustic Characterization of Dog Vocalizations

In any classification problem, we need to identify, define and explain the attributes that will characterize the samples presented to the learning algorithm. In this section, we describe the information used to acoustically characterize all the audio samples (the 10-second-long samples employed in the playback test and the manually classified segments) to process them automatically using machine learning techniques that help us to build classification models. We used the openSMILE software to extract Low-Level Descriptors (LLDs). We extracted the LLDs included in the large openSMILE emotion feature. The LLDs were computed using a frame size of 25 ms and a frame step of 10 ms. A moving average filter for smoothing data contours was applied. To have the same number of attributes for each single vocalization recording, regardless of its duration, it was computed 39 statistical functions over the values of the LLDs and their deltas and double deltas coefficients in each frame of the recording. We obtained a total of 6,552 attributes for each single audio sample. After an experimentation stage, we decided to use the *Relief Attribute* feature selection method as implemented in Weka. This method showed the best accuracy rates when we took the 350 best-ranked attributes. We selected these features from the original set of 6,552 features to obtain the best ones and reduce the dimensionality of the features vector.

2.5 Classification and Validation Methods

We used the acoustic characterization described in the section 2.4 and applied the machine learning technique *Support Vector Machines* (SVM) using a polynomial kernel to classify different dog vocalizations and various types of barks. We selected SVM given that this technique has been successfully used to classify human emotions employing a similar acoustic feature set [18]. Unlike the assessment presented in section 2.3, where we based the analysis on the human evaluations, in this section, it is done using automated tools. This analysis allows an objective assessment of the data. To measure the impact and dependency on individuals in the classification, we validated the models by two methods:

10 Fold Cross Validation (10FCV) In this validation scheme, a classification model is trained using the 90% of the samples in the dataset. We tested it with the 10% left out. This validation round is repeated ten times, each time leaving out a different set of samples. We used this validation scheme to have a baseline accuracy. However, given that we extracted several samples from the same recording, they could generate an effect of pseudo-replication.

Leave One Dog Out Validation (LODOV) We implemented a scheme of cross validation where a classification model is trained using all the samples of N-1 dogs and tested it with the one left out. N is the total number of dogs in the data set, 23 dogs in our case. This validation round is repeated N times, each time leaving out a different dog. This scheme ensures that the trained model is dog-independent. We calculated the accuracy statistics from the accumulated confusion matrix.

3 Results

3.1 Generated Database

This database contains almost eleven hours of audio and video recordings of two breeds Chihuahua and Schnauzer. This database could be an interesting resource from the perspective of the computer science and also from the point of view of the animal behavior research areas. There is currently a trend towards the study of the application of computational systems in the interaction with dogs [13, 30, 24, 12]. In this research context, a bark collection like the generated in this work could be interesting in order to explore the dog-computer interaction through interfaces that take barks as input.

3.2 Agreement of the Participating Judges

It is important to evaluate the agreement among the assessments of the judges because a good agreement means that the emotions primitives expressed in dog's vocalization are perceived similarly by humans. Therefore, these emotions primitives are well suited for the modeling of dog's emotions. We calculated an agreement measurement among judges when they evaluated the playback material. We used the Cronbach's alpha coefficient [7] that is widely used to describe the reliability of tests and evaluations when there is no missing data (as in our case). We obtained a good agreement among the assessments of the judges. Table 2 shows the absolute agreement among the answers of the judges. We can see that, in general, we obtained a good agreement.

Table 2. Absolute agreement between the answers of participants measured using alpha coefficients for both primitives and both breeds. The fourth column shows the interpretation of the alpha coefficient.

Breed	*Dimension*	*Alpha*	*Reliability*
Chihuahua	Arousal	0.838	Good
	Valence	0.858	Good
Schnauzer	Arousal	0.869	Good
	Valence	0.775	Acceptable

3.3 Statistical Analysis

We performed statistical tests to the data obtained by the subjective assessment of recordings. To complete this analysis, we used the software *IBM SPSS Statistics Version 24*. We applied a Spearman's rho test to the combination of variables to detect a correlation between them. An interesting research question is how strong is the correlation between valence and arousal. As in the case of human emotions, this two-dimensional representation, on average, present a weak but consistent V-shaped relation of arousal as a function of valence [15]. The more negative or positive emotions usually rated to have higher arousal level too, while more neutral ones are low on arousal too. It is important because these variables are two emotional dimensions that must be weakly correlated to map a variety of emotions. We can see that the Spearman's rho correlation between these two variables is -0.044. This result indicates that there is a very weak correlation between the perception of both primitives. Therefore, it is possible to use them as emotion descriptors. We used the non-parametric test chi-square to measure the dependency between the nominal variables: emotion primitives and type of vocalization. We found a significant statistical association between the vocalizations of dogs (aggressive bark, very aggressive bark, growl, and whine) and the value of arousal annotated by the participants. We identified that judges perceived growls as very negative and active; and whines as negative and very passive. We also found a significant correlation (0.347) between these same variables. The comparison between vocalization and valence also shown a significant association and a significant correlation (0.431). This result indicates that judges perceived differences in the emotion primitives when they listened to a different category of dog vocalization (*i. e.*, the perceived emotions changed according to the type of vocalization). Aggressive barks were evaluated mainly as a very negative and active vocalization. Very few judges evaluated aggressive barks as very positive and very passive vocalizations. This fact confirms that the proposed scale makes sense for the assessment of emotions in dog vocalizations.

3.4 Automatic Classification of Dog Vocalizations

The tables 3 and 4 show the performance of the classification experiments in terms of precision, recall and F-measure. The F-measure is metric for classification performance that is calculated as the harmonic mean of precision and recall. The F-measure score reaches its best value at 1 when precision and recall are 1, and worst at 0 when precision or recall is 0 (*i.e.*, the closest to 1, the most accurate is the classification). As we can see in Table 3, we obtained good results classifying the three different vocalizations and contrasting them against to other domestic noises captured during the recordings.

Table 3. Automatic classification of dog vocalizations.

Class	***Precision***	***Recall***	***F-Meassure***
Whine	0.899	0.856	0.877
Growl	0.907	0.803	0.852
Bark	0.951	0.950	0.950
Sound	0.933	0.940	0.937
Weighted Avg.	**0.942**	**0.942**	**0.942**

We included the domestic sounds category because it is relevant to build real-world applications based on automatic dog's vocalization recognition. When the classification system works in realistic environments, it will have to discriminate between dog's vocalizations and other sounds before classifying types of vocalizations. Table 4 shows the results of classifying only barks according to the emotional state. In this case, we have four classes: aggressive, very aggressive, playfulness and other (barks not generated by the stimuli tasks). The playfulness barks have the best F-measure. We applied two validation schemes to measure the impact of individual dog acoustic properties in the modeling of emotion recognition. We observed a significant decrease in recognition performance when we applied a LODOV, from an average F-measure of 0.79 to 0.49. This fact tells us that each dog has particularities in the way it vocalizes. It implies that recognition models should include a way to adapt to the individual peculiarities to maintain an acceptable accuracy.

3.5 Automatic Assessment of Emotion Primitives in Dog Vocalizations

Finally, in this section, we compare the classification made by the system and the classification made by the human judges. We trained regression models with

Table 4. Automatic classification of dog vocalizations (A) *stratified 10-FCV*, and (B) *leave one speaker out* validation schemes.

Response	***Precision***		***Recall***		***F-Measure***	
	A	B	A	B	A	B
Very aggressive	0.75	0.27	0.62	0.16	0.68	0.20
Aggressive	0.79	0.59	0.87	0.73	0.83	0.65
Playful	1.00	0.75	0.99	0.98	0.99	0.85
Other	0.82	0.27	0.78	0.18	0.80	0.21
Weighted Avg.	**0.79**	**0.48**	**0.79**	**0.53**	**0.79**	**0.49**

the audio samples used in the subjective assessment of emotion primitives. The trained models automatically estimate a continuous value in the same range of emotion primitives than the training samples. In this case from 1 to 5 that are the possible values of arousal and valence given by the judges using the SAMS scale. We used the objective information provided by the acoustic features described in section 2.4 and the algorithm SMOreg *Support Vector Machines for Regression*. We validated by 10FCV. We found that it is possible to train a pattern recognition model to recognize emotions in dog vocalizations with a good estimation performance. For the primitive valence, we obtained a Pearson's correlation coefficient of 0.6341, mean absolute error 0.3031 and root mean squared error 0.3031. For the primitive arousal, we obtained a Pearson's correlation coefficient of 0.7048, mean absolute error 0.3788 and root mean squared error 0.4782. These results are very encouraging and confirm the viability of building automatic systems for dog emotion recognition.

4 Discussion

In the previous sections, we presented a methodology to create a database of emotional dog vocalizations. We used the collected data to validate a method for emotion assessment on dog vocalizations based on subjective assessments by human listeners. We trained pattern recognition models to identify dog vocalizations from other domestic sounds, to classify barks by the context of inductions and to estimate the emotional content of dog vocalizations. The obtained results showed that the primitives arousal and valence are pertinent to assess dog's emotions given that humans correctly perceived emotion information using this bi-dimensional model. These primitives revealed to be independent of between them across all dogs, which allows us to map a variety of emotions. We also identified that automatic emotion recognition in vocalizations is affected by individual acoustic patterns. Inducing and recording barks in dogs is a difficult task. We were able to obtain a lot of data during the aggression to unfamiliar people stimuli but few during the playing sessions.

The presence of the experimenter seems to affect the habitual dog's behavior when the owner tries to play with them. The perceived emotion primitives reflect this situation. The percentage of negative and very negative vocalizations was greater than the proportion of positive and very positive vocalizations. On the other hand, it is hard to obtain very passive vocalizations given that aggression and playfulness are mainly active and very-active emotions. It is hard to obtain 10-second long segments for all the vocalizations in the database to carry on the emotion evaluation of vocalizations. For this reason, did not include all the types of vocalizations in the human assessment of emotions. The participants in the emotional evaluation of dog vocalizations felt comfortable with the graphical five-points emotion scale that we designed. This instrument was beneficial for our purposes. We found, in general, good agreement between the answers of participants in the emotional evaluation of dog vocalizations.

Arousal had a better agreement than valence in Chihuahua dogs. On the other side, valence had a better agreement than arousal for Schnauzer dogs. We also found that this task is very subjective. The individual reliability of a single participant was poor in all cases. Valence and arousal showed a weak correlation. However, we identified that many barks with active and very active arousal have very negative valence. We identified that judges perceived growls as very negative and active; and whines as negative and very passive. Also, they recognized aggressive barks as very negative and active. Unfortunately, we did not have enough playfulness samples to include in the emotion evaluation study. Finally, we were able to train accurate models to recognize and classify dog vocalizations automatically.

We also trained accurate models to classify the different types of barks. We found that classification models for dog vocalizations could be dog dependent. When we used a scheme validation that leaves out one dog from the training samples, the classification performance decreased drastically. Even though the playfulness class had few samples, it was the most recognizable type of barks. The classes very aggressive and aggressive had a high level of confusion between them. The classifier accurately differentiated the dogs' vocalizations from other domestic sounds. The pattern recognition models that we developed obtained an acceptable accuracy in the estimation of emotion primitives. Similarly to other works [25, 19], we found that estimating valence from acoustic features is harder than estimating arousal. The methodology presented can be a significant contribution to the automatic analysis of dog's barks. The machine learning approach linked with the human assessment of emotions is a novel and interesting methodology that can open the way for new opportunities in the applied field and also can have high commercial interest too.

5 Conclusions

In this work, through a subjective evaluation of recorded vocalizations, we observed that human judges correctly perceived variations of primitive emotional vocalizations of different dogs. We conducted experiments using machine learning techniques with which we found a particular way of expressing emotions vocally by each dog. Finally, we trained a model for acoustic pattern recognition aimed at estimating emotional primitives in dog vocalizations. The performance obtained by this method suggests that it is possible to build computer systems able to assess the emotional state in dogs based on their vocalizations. Using this technology, developers could design applications oriented to the welfare of dogs, for example, alerting the owners when the dog shows a constant negative emotional state and/or low arousal levels. Other applications could be for domestic security, for example, an app that sends an alert to the owners when they are not at home (or activate a security mechanism), informing that barks with high arousal and negative valence.

Ethical Considerations

Our research is based on non-invasive procedures for evaluating dog behaviour. During recordings we observed that dogs live in optimal health conditions with food and water ad libitum. The owners signed a consent form to voluntarily permit their dogs participate in the present study, and that the resulting media can be used in publications.

Acknowledgements. This work has been carried out in the context of the "Cátedras CONACyT" program funded by the Mexican National Research Council (CONACyT). We thank to Doris Contreras, Fernando Flores, Kenia Rivera, Hector Cruz, Dulce Nolasco and Abraham Galicia for the assistance with the building of the database. This work was partially funded by CONACYT under the Thematic Networks program (Language Technologies Thematic Network project 281795).

References

1. Banse, R., Scherer, K.R.: Acoustic profiles in vocal emotion expression. Journal of personality and social psychology 70(3), 614 (1996)
2. Bradley, M.M., Lang, P.J.: Measuring emotion: the self-assessment manikin and the semantic differential. Journal of behavior therapy and experimental psychiatry 25(1), 49–59 (1994)
3. Briefer, E.: Vocal expression of emotions in mammals: mechanisms of production and evidence. Journal of Zoology 288(1), 1–20 (2012)
4. Briefer, E.F., Tettamanti, F., McElligott, A.G.: Emotions in goats: mapping physiological, behavioural and vocal profiles. Animal Behaviour 99, 131–143 (2015)
5. Burgdorf, J., Panksepp, J., Moskal, J.R.: Frequency-modulated 50khz ultrasonic vocalizations: A tool for uncovering the molecular substrates of positive affect. Neuroscience & Biobehavioral Reviews 35(9), 1831–1836 (2011)
6. Cowie, R., Cornelius, R.R.: Describing the emotional states that are expressed in speech. Speech communication 40(1), 5–32 (2003)
7. Cronbach, L.J.: Coefficient alpha and the internal structure of tests. psychometrika 16(3), 297–334 (1951)
8. Faragó, T., Andics, A., Devecseri, V., Kis, A., Gácsi, M., Miklósi, Á.: Humans rely on the same rules to assess emotional valence and intensity in conspecific and dog vocalizations. Biology letters 10(1), 20130926 (2014)

9. Faragó, T., Pongrácz, P., Range, F., Virányi, Z., Miklósi, Á.: The bone is mine: affective and referential aspects of dog growls. Animal Behaviour 79(4), 917–925 (2010)
10. Feddersen-Petersen, D.: Vocalization of european wolves (canis lupus lupus l.) and various dog breeds (canis lupus f. fam.). Archiv fur Tierzucht 43(4), 387–398 (2000)
11. Grimm, M., Mower, E., Kroschel, K., Narayanan, S.: Combining categorical and primitives-based emotion recognition. In: Signal Processing Conference, 2006 14th European. pp. 1–5. IEEE (2006)
12. Hirskyj-Douglas, I., Read, J.: The ethics of how to work with dogs in animal computer interaction. In: Proceedings of the Animal Computer Interaction Symposium. Measuring Behaviour (2016)
13. Hirskyj-Douglas, I., Read, J.C.: Who is really in the center of dog computer design? In: Proceedings of the 2014 Workshops on Advances in Computer Entertainment Conference. p. 2. ACM (2014)
14. Knutson, B., Burgdorf, J., Panksepp, J.: Ultrasonic vocalizations as indices of affective states in rats. Psychological bulletin 128(6), 961 (2002)
15. Kuppens, P., Tuerlinckx, F., Russell, J.A., Barrett, L.F.: The relation between valence and arousal in subjective experience. Psychological Bulletin 139(4), 917 (2013)
16. Larranaga, A., Bielza, C., Pongrácz, P., Faragó, T., Bálint, A., Larranaga, P.: Comparing supervised learning methods for classifying sex, age, context and individual mudi dogs from barking. Animal cognition 18(2), 405–421 (2015)
17. Molnár, C., Kaplan, F., Roy, P., Pachet, F., Pongrácz, P., Dóka, A., Miklósi, Á.: Classification of dog barks: a machine learning approach. Animal Cognition 11(3), 389–400 (2008)
18. Pérez-Espinosa, H., Pérez-Martınez, J.M., Durán-Reynoso, J.Á., Reyes-Meza, V.: Automatic classification of context in induced barking. Research in Computing Science 100, 63–74 (2015)
19. Pérez-Espinosa, H., Reyes-García, C.A., Villaseñor-Pineda, L.: Acoustic feature selection and classification of emotions in speech using a 3d continuous emotion model. Biomedical Signal Processing and Control 7(1), 79–87 (2012)
20. Pollermann, B.Z., Archinard, M.: Acoustic patterns of emotions. Improvements in speech synthesis p. 237 (2002)
21. Pongrácz, P., Molnár, C., Dóka, A., Miklósi, Á.: Do children understand man's best friend? classification of dog barks by pre-adolescents and adults. Applied animal behaviour science 135(1), 95–102 (2011)
22. Pongrácz, P., Molnár, C., Miklósi, Á.: Barking in family dogs: an ethological approach. The Veterinary Journal 183(2), 141–147 (2010)
23. Pongrácz, P., Molnár, C., Miklósi, A., Csányi, V.: Human listeners are able to classify dog (canis familiaris) barks recorded in different situations. Journal of Comparative Psychology 119(2), 136 (2005)
24. Pons, P., Jaen, J., Catala, A.: Envisioning future playful interactive environments for animals. In: More Playful User Interfaces, pp. 121–150. Springer (2015)
25. Schuller, B., Vlasenko, B., Eyben, F., Rigoll, G., Wendemuth, A.: Acoustic emotion recognition: A benchmark comparison of performances. In: Automatic Speech Recognition & Understanding, 2009. ASRU 2009. IEEE Workshop on. pp. 552–557. IEEE (2009)
26. Svartberg, K., Forkman, B.: Personality traits in the domestic dog (canis familiaris). Applied animal behaviour science 79(2), 133–155 (2002)
27. Wu, D., Parsons, T.D., Narayanan, S.S.: Acoustic feature analysis in speech emotion primitives estimation. In: INTERSPEECH. pp. 785–788 (2010)

28. Yin, S.: A new perspective on barking in dogs (canis familaris.). Journal of Comparative Psychology 116(2), 189 (2002)
29. Yin, S., McCowan, B.: Barking in domestic dogs: context specificity and individual identification. Animal Behaviour 68(2), 343–355 (2004)
30. Zeagler, C., Gilliland, S., Freil, L., Starner, T., Jackson, M.: Going to the dogs: towards an interactive touchscreen interface for working dogs. In: Proceedings of the 27th annual ACM symposium on User interface software and technology. pp. 497–507. ACM (2014)

ISSN 1870-4069

Social Media Network Simplification Using Morphological Operators

Erick López-Ornelas, Rocío Abascal Mena

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

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

Abstract. Complex networks arise in many diverse contexts, ranging from web pages and their links, computer networks, and social networks interactions. The modelling and mining of these large-scale, self-organizing systems is a broad effort spanning many disciplines. This article proposes the use of morphological operators, based on Mathematical Morphology, to simplify a set of interactions in a complex social media network. By applying these morphological operators, it is possible to simplify the social network and thus identify important communities and actors in the network. An analysis based on the visualization of the communities was carried out to verify the pertinence of the detection and simplification.

Keywords. Morphological operators, mathematical morphology, social media network, network simplification.

1 Introduction

Social media networks such as microblogs and social networks e.g. Twitter, LinkedIn and Facebook, are provide interactive and cheaper way for user to share ideas, exchange information and stay connected with people. Ease in using social media applications on mobile devices achieves rapid growth in social media network users and leads to generate vast amount of user generated content.

This large user base and their discussions produces huge amount of user generated data. Such social media data comprises rich source of information which is able to provide tremendous opportunities for companies to effectively reach out to a large number of audience.

With the current popularity of these Social media networks (SMN), there is an increasing interest in their measurement and modelling. In addition to other complex networks properties, SMN exhibit shrinking distances over time, increasing average degree, and bad spectral expansion.

Unlike other complex networks such as the web graph, models for SMN are relatively new and lesser known. In this kind of networks, models may help detect, simplify and classify communities, and better clarify how news and gossip is spread in social networks.

Network simplification can provide benefits to applications of various domains and for suggesting like-minded people to user which are still unknown to him/her.

An important practical problem in social networks is to simplify the network of users based on their shared content and relationship with other users. Community in network is a pattern with dense links internally and sparse links externally. These links can be characterized by the content similarity between users, friendship between them and also other similarities in their personal data such as their location, gender, age etc.

In other hand, Mathematical morphology is generally studied as an aspect of image processing [1]. As digital images are usually two-dimensional arrangements of pixels, where spatial relationships between elements of the image are essential features.

Mathematical Morphology is a theory that studies the decomposition of lattice operators in terms of some families of elementary lattice operators [2]. When the lattices are considered as a multidimensional graph (e.g. Social Media Network), the elementary operators can be characterized by structuring functions. The representation of structuring functions by neighborhood graphs is a powerful model for the construction of morphological operators.

This article proposes the use of morphological operators, based on Mathematical Morphology, to simplify a set of interactions in a complex social network. By applying these morphological operators, it is possible to simplify the social network and thus execute important queries in the network.

The structure of the article is as follows. In Section 2 the materials and methods are shown giving the essentials of mathematical morphology and how this technique is applied on SMN. The essential operators applied are shown and explained (Opening and Closing). An example of this simplification is shown in section 3 and the query modeling is explained. The conclusions and directions for further work are given in the final section.

1.1 Complex Networks Properties

Researchers are now in the position of observing how SMN evolve over time, and how the network is constructed. Unlike in traditional social network analysis, social interaction of millions of people from across the world can be elicited and some properties can be described. For definitions of the terms used below, see [3].

(i) Large-scale. SMN are examples of complex networks with number of nodes often in the millions; further, some users have disproportionately high degrees. About half a billion users are registered on Facebook [4]. Some of the nodes of Twitter corresponding to well-known celebrities including Katy Perry, Justin Bieber or Barak Obama have degree over ninety million [5].

(ii) Small world property and shrinking distances. The small world property, is a central notion in the study of complex networks [6]. The small world property demands a low diameter and a higher clustering coefficient than can be found in this kind of network [7]. Low diameter and high clustering coefficient were reported in the Twitter by Java et al. [8].

(iii) Power law degree distributions. Power laws were observed over a decade ago in subgraphs sampled from the web graph and are ubiquitous properties of complex networks. Kumar [9] studied the evolution of Flickr and found that this network exhibit power-law degree distributions.

(iv) Bad spectral expansion. Social networks often organize into separate clusters in which the intra-cluster links are significantly higher than the number of intercluster links. In particular, social networks contain communities, where some groups correspond to the clusters [10]. As a result, this kind of network possess bad spectral expansion properties realized by small gaps between the first and second eigenvalues of their adjacency matrices.

1.2 Some Techniques of Network Simplification

Similar work has been conducted on simplifying networks. In [11], the authors developed 3 different algorithms. The first decomposes a large network into some smaller sub-networks, generally overlapped. The remaining two carry out simplification based on commute times within the network. The algorithms produce a multilayered representation. All three algorithms use their simplified representations to perform matching between the input network.

In [12], the author uses a simplification algorithm to generate simplified network for input into their network layout algorithms. The network is not visualized and presented to the user as a way to help them better understand the network. Instead, a series of progressively simplified networks are used to guide the positioning of the nodes in the network.

Additional simplification algorithms have been proposed to assist in robot path planning [13], classifying the topology of surfaces [14], and improving the computational complexity and memory requirements of dense graph processing algorithms [15].

Simplification may also be accomplished through the clustering technique. In [16], authors define one such clustering algorithm for better visualizing the community structure of network graphs. The kind of graphs that they are targeting with this technique are those that would have a naturally occurring community structure.

Some authors present another approach to visually simplifying large scale graphs [17]. They have developed different methods for randomly sampling a network and using that sampling to construct the visual representation of the complex network.

In [18], the authors focus on providing metrics for simplifying graphs that represent specific network topologies. The goal of this work is to simplify and visualize complex network graphs while maintaining their semantic structures. Although network topologies are certainly reasonable candidates for these visualization techniques, there are other sets of graph data that could greatly benefit from simplification techniques for visualization. Their general approach is similar to ours in that they are using characteristics of the graphs that frequently occur, using Morphological operators some physical characteristics are kept and some nodes "grow" and some "shrink" in order to obtain a better simplification. This process will be explained in next section.

2 Materials and Methods

2.1 Mathematical Morphology and Networks

The study on Mathematical morphology, started at the end of the Sixties and was proposed by Matheron and Serra [19]. Mathematical morphology rests in the study of the geometry and forms; the principal characteristic of the morphological operations is image segmentation and conservation of the principal features forms of the nodes [1].

Despite its origin, it was recognized that the roots of this theory were in algebraic theory, notably the framework of complete lattices [20]. This allows the theory to be completely adaptable to non-continuous spaces, such as graphs and networks. For a survey of the state of the art in mathematical morphology, we recommend [21].

2.2 Social Media Network Representation

SMN is a graph representation having a set of data where some pairs of data are connected by links. Once a SMN representation is adopted, the interconnected data are called vertices or nodes of the Network and the links that connect vertices are called edges or arcs. An edge of the graph is then simply a pair of connected vertices. Thus, a network is made of a set of vertices and of a set of edges. We can also associate to each vertex and/or to each edge a weight that represents some kind of measure on the network (in our case friends, likes, follows, mentions, etc.), leading to weighted graphs.

In terms of notation, Social Media networks employ the typical graph notation $G = (V, E)$, where G stands for the whole network, V stands for the set of all vertices and E for the set of all edges. Due to the different types of vertices and edges in such networks, it is common to consider sets of vertices and edges within V and E that contain vertices and edges of the same type. For instance, in the case of a photo sharing and tagging network, one can consider the set of vertices V to comprise the users, photos and tags of the system, i.e. $V = \{U, P, T\}$. Similarly, the set of edges in such an application would comprise the set of user-photo, photo-tag and user-tag associations, $E= \{UP, PT, UT\}$.

This graph (V,E) will have the following characteristics: (i) it is "non-oriented", that is: $\forall(v,w) \in E, (w,v) \in E$; (ii) it is 1-graph: the same pair cannot be represented more than one time in the family of arcs E; (iii) it has no loops: $\forall v \in V, (v,v) \notin E$.

2.3 Morphological Operators

The algebraic basis of Mathematical morphology is the lattice structure and the morphological operators act on lattices [2]. In other words, the morphological operators map the elements of a first lattice to the elements of second one (which is not always the same as the first one). A lattice is a partially ordered set such that for any family of elements, we can always find a least upper bound and a greatest lower bound (called a supremum and an infimum). The supremum (resp., infimum) of a family of elements is then the smallest (greatest) element among all elements greater (smaller) than every element in the considered family.

The supremum is given by the union and the infimum by the intersection. A morphological operator is then a mapping that associates to any subset of nodes another subset of nodes. Similarly, given a graph, one can consider the lattice of all sub sets of vertices [22] and the lattice of all subsets of edges. The supremum and infimum in these lattices are also the union and intersection. In some cases, it also interesting to consider a lattice whose elements are graphs, so that the inputs and outputs of the operators are graphs.

The algebraic framework of morphology relies mostly on a relation between operators called adjunction [2]. This relation is particularly interesting, because it extends single operators to a whole family of other interesting operators: having a dilation (resp., an erosion), an (adjunct) erosion (resp., a dilation) can always be derived, then by applying successively these two adjunct operators a closing and an opening are obtained in turn (depending which of the two operators is first applied), and finally composing this opening and closing leads to alternating filters.

Firstly, they are all increasing, meaning that if we have two ordered elements, then the results of the operator applied to these elements are also ordered, so the morphological operators preserve order. Additionally, the following important properties hold true:

- the dilation (resp., erosion) commutes under supremum (resp., infimum);
- the opening, closing and alternating filters are indeed morphological filters, which means that they are both increasing and idempotent (after applying a filter to an element of the lattice, applying it again does not change the result);
- the closing (resp., opening) is extensive (resp., antiextensive), which means that the result of the operator is always larger (resp., smaller) than the initial object;

In the graph *G (V, E),* if the vertex (V_i) of the graph constitutes the digital grid and its neighbors their interactions, then the process compares and affects the interaction value of v_i on the graph constructed using the morphological transformations. These transformations are the core of the simplification.

The principle of the growing/shrinking the graph consists in transform the *G(v)* value by affecting the nearest interaction value $val(v_i)$ present among the *v* neighbor's nodes. The new graph v_n is then the result of the fusion of nodes. To carry out this transformation, the morphological operations on the graph are applied and a loop is generated until the reach of one threshold parameter.

Let us assume that we have a flat structuring element that corresponds to the neighbor's nodes Structuring Element (*SE≡NE(v)*). Then the eroded graph *ε(G(v))* is defined by the infimum of the values of the function in the neighborhood [23]:

$$\varepsilon(G(v)) = \{\wedge\ G(v_i),\ v_i \in N_E(v)\ \cup \{v\}\}.$$

Dilation δ(G(v)) is similarly defined by the supremum of the neighboring values and the value of *G(v)* as:

$$\delta(G(v)) = \{\vee G(v_i), v_i \in N_E(v) \cup \{v\}\}.$$

Classically, opening *γ* is defined as the result of erosion followed by dilation using the same SE:

$$\gamma(G(v)) = \delta(\varepsilon(G(v))).$$

Similarly, closing *φ* is defined as the result of dilation followed by erosion with the same SE:

$$\varphi\ (G(v)) = \varepsilon\ (\delta(G(v))).$$

The geometrical action of the openings and closings transformations, *γ(G(v))* and *φ(G(v))* respectively, produce a growing/shrinking of the graph. Of course, this fusion process can be regulated using parameters for the opening and closing, but also we can regulate the growing depending on the information that we need to compare. The graph has to be updated to keep aggregating the different nodes always applying the morphological transformations of *γ(G(v))* and *φ(G(v))* until their parameter value is reached. In figure 1 and figure 2, some morphological operations are shown. We can see the difference applying different morphological operators on the same graph.

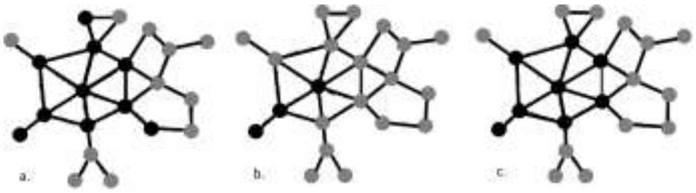


Fig. 1. a. Random graph selected. b. Eroded *ε(G(v))*. c. Dilated *graph δ(G(v))*.

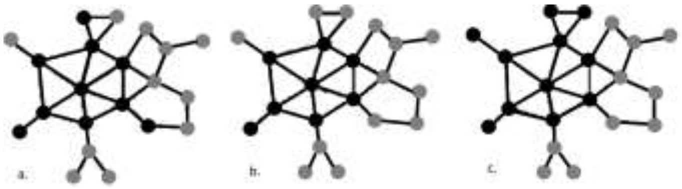


Fig. 2. a. Random graph selected. b. Opening transformation *γ(G(v)).* c. Closing transformation *φ (G(v)).*

3 Experiment

3.1 Social Network Simplification

In this article, as an experiment, we show a set of interactions on Twitter. This information was extracted from Twitter and explores a trend topic appeared in México. The hashtag is *#estafamaestra*, and it was arising from the corruption scandals generated in Mexico in September 2017. Among the multiple elements of analysis, we decided to use morphological operators in order to simplify the original network, the study was made taking into account the characteristics of each node for their simplification.

The information concentrated in figure 3 corresponds to the extraction carried out on September 5, 2017. In the network are represented 3399 nodes (tweeters) and 5502 arcs (interactions) that were made between them. This is a complex interaction network, so it is important to simplify it in order to perform a better analysis of the interactions that were generated in the social network.

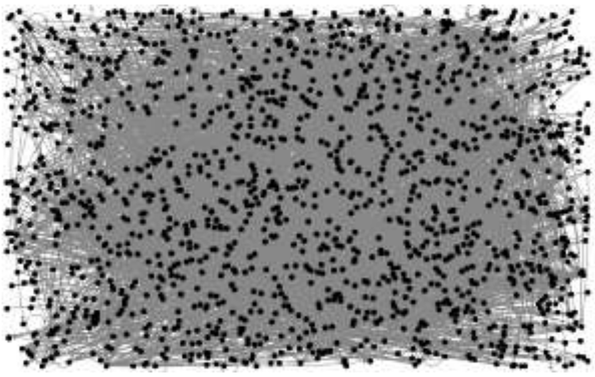


Fig. 3. Complete random graph selected with 3399 nodes and 5502 interactions.

3.2 Social Media Network Representation

At the most abstract level, given a Social Media network *G = (V, E)*, where *G* stands for the whole network, *V* stands for the set of all vertices and *E* for the set of all edges, each Social Media interaction can be defined as a subgraph of the network comprising a set $V_C \subseteq V$ of Social Media entities that are associated with a common element of interest.

This element can be as varied as a topic, a real-world person, a place, an event, an activity or a cause.

For instance, in the case of Twitter network, one can consider the set of vertices *V* to comprise the users, mentions, tweet content, tweet favorites and retweets, i.e. *V = {U, M, Tc, Tf, Rt}*. The edges in such an application would comprise the set of followed, followers, tweet number, image profile and location, *E= {Fd, Fs, Tn, Ip, L}*.

Even if we can use all these characteristics to apply morphological operators, we have decided to only use 3 node elements to carry out the simplification. These elements are the mentions, number of favorites and retweets represented by *V = (M, Tf, Rt).*

3.3 Decisional Node Aggregation

The principle of the union of nodes consists in transform the *G(v)* value by affecting the nearest *Tf* value $val(v_i)$ present among the *v* neighbors, and the grouping process is the union of nodes $(v_i \cup v_j = v_n)$. The new node v_n is then the result of the fusion of nodes. To carry out this transformation, the morphological operations on the graph are applied.

Let us assume that we have a flat structuring element that corresponds to the neighborhood Structuring Element $(SE \equiv N_E(v))$. Then the eroded graph $\varepsilon(G(v))$ *is* defined by the infimum of the values of the function in the neighborhood and represents the minimum value found on the neighbors [24]:

$$\varepsilon(G(v)) = min\,\{G(v_i),\, v_i \in N_E(v) \cup \{v\}\}.$$

Dilation $\delta(G(v))$ is similarly defined by the supremum of the neighboring values and the value of $G(v)$ and it is represented by the maximum value found on the neighbors as:

$$\delta(G(v)) = max\{G(v_i),\, v_i \in N_E(v) \cup \{v\}\}.$$

Classically, opening γ is defined as the result of erosion followed by dilation using the same SE:

$$\gamma(G(v)) = \delta(\varepsilon(G(v))).$$

Similarly, closing φ is defined as the result of dilation followed by erosion with the same SE:

$$\varphi\ (G(v)) = \varepsilon\ (\delta(G(v))).$$

The geometrical action of the openings and closings transformations, $\gamma(G(v))$ and $\varphi(G(v))$ respectively, produce a growing or shrinking of the selected graph. Of course, this fusion process can be regulated using parameters for the opening and closing, but also we can regulate the fusion depending on the mentions, tweet favorites or retweets. The graph has to be updated to keep aggregating the different nodes always applying the morphological transformations of $\gamma(G(v))$ and $\varphi(G(v))$ until their parameter value is reached.

For merging two adjacent nodes in a graph, certain V conditions should be verified. We can define some mention parameters that condition the difference between these values of two adjacent nodes that can be aggregate at the opening and closing operations. These parameters are called the minimal mention parameter d_1 and the maximal mention threshold d_2. To use them, we should calculate, in a first time, the mention differences in the graph. So, we calculate $d_1(G(V_i), max(G(V)))$, the difference between the maximum value of mentions in the neighboring nodes, and $d_2(G(V_i), min(G(V)))$ the minimal difference. If the maximal mention parameter is higher than d_1, the opening operation $\gamma\ (G(v_i))$ does not merge nodes. Also, if the minimal mention parameter is higher than d_2, the closing operation $\varphi\ (G(V_i))$ does not merge nodes. A loop is he required to perform all the necessary aggregations for the simplification of the graph. In figures 4-9 we show the simplification process in different steps.

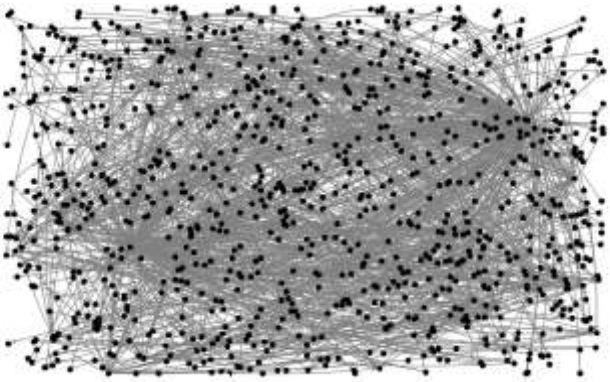


Fig. 4. Graph iterations = 5 Morphological operations applied = γ and φ, $d_1 = 50$, $d_2 = 30$, Nodes = 1535 , Interactions = 2045.

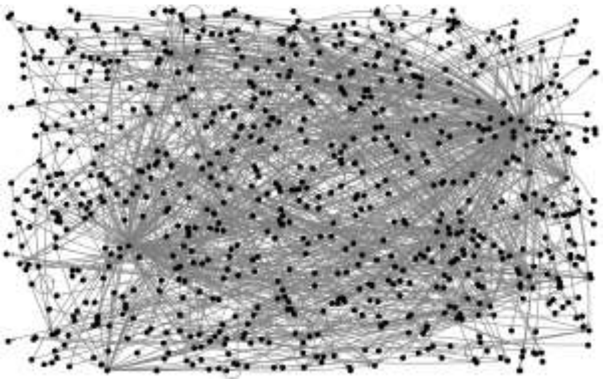


Fig. 5. Graph iterations = 20 Morphological operations applied = γ and φ, $d_1 = 50$, $d_2 = 30$, Nodes = 1381, Interactions = 1715.

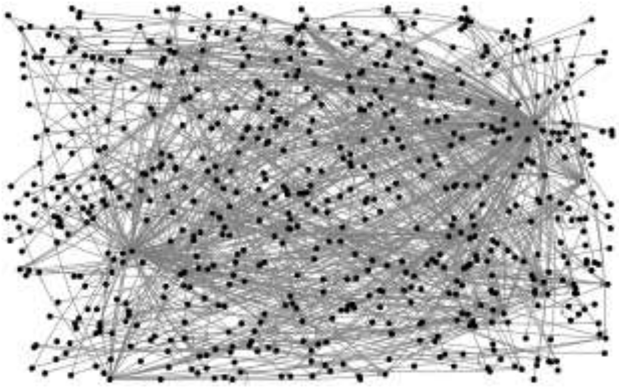


Fig. 6. Graph iterations = 35 Morphological operations applied = γ and φ, $d_1 = 50$, $d_2 = 30$, Nodes = 925, Interactions = 1251.

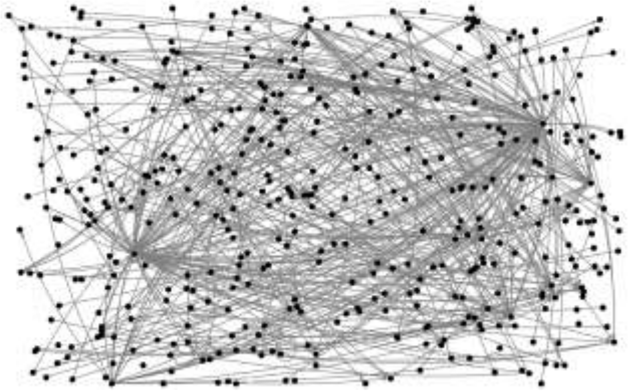


Fig. 7. Graph iterations = 50 Morphological operations applied = γ and φ, $d_1 = 50$, $d_2 = 30$, Nodes = 509, Interactions = 563.

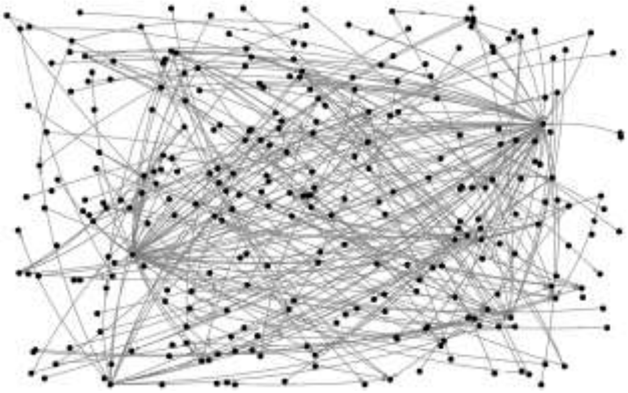


Fig. 8. Graph iterations = 65 Morphological operations applied = γ and φ, $d_1 = 50$, $d_2 = 30$, Nodes = 221, Interactions = 330.

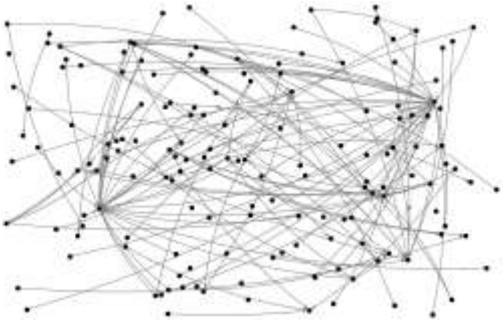


Fig. 9. Final graph. Graph iterations = 80 Morphological operations applied = γ and φ, $d_1 = 50$, $d_2 = 30$, Nodes = 118 , Interactions = 105.

In figure 10, we show the simplification rate when the morphological operators are applied. It is interesting to note that simplification is more significant in the first iterations, usually in the first 5 iterations, which is normal regarding the parameters d_1 and d_2 used. Then, the parameters do not cause so much effect and the simplification rate remains stable. It is at that time that the iterations and the use of morphological operators must stop.

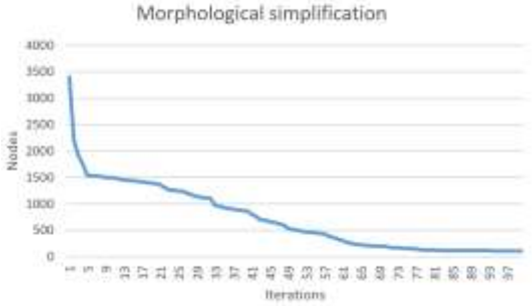


Fig. 10. Simplification rate using morphological operators.

The computation time needed to perform simplification is not significant. Each graph simplification having 5000 nodes requires about 5 seconds to apply morphological operations and compute the entire network. Even if the parameters chosen d_1 and d_2 were widely dispersed, the time is up to 15 seconds to complete the computation process on an Intel Core i9.

3.4 Network Information Retrieval

The final node characteristics is calculated using the final graph after the use of the morphological transformations of $\gamma(G(v))$ and $\varphi(G(v))$. These characteristics $\{C\}$ are then stored separately in a database, which is useful to make meanly queries.

These features $\{C\}$ called “metadata” [25, 26] characterizing each node are then stored and handled separately.

There are two different features extracted from the graph: (i) “node properties”, that are specific to each node (user name, friends, followed, followers, tweet number, image profile and location, etc.) and (ii) “interaction characteristics”, that describe the tweet (mentions, tweet content, tweet favorites and retweets).

The graph G is then considered as a set of layers information $\{L\}=(l_1,\ldots,l_n)$. Each layer is associated with a set of characteristics $\{C\}$, with their values $\{v\}$, which defines the structure of each instance stored according to the layer. Each instance is identified by a unique node identifier O_{id} and a descriptor called the value $v(v=(v_1,\ldots, v_n))$.

To retrieve information from the final graph we decided to use *Cypher* [27] that is a declarative graph query language that allows for expressive and efficient querying and updating of the graph store. *Cypher* is a simple but powerful query language. This language allows you to focus on the domain instead of getting lost in graph database access.

Being a declarative language, *Cypher* focuses on the clarity of expressing "what" to retrieve from a graph, not on "how" to retrieve it.

Retrieving information is made through a rule expressed by a query [28]. Defining a query relies on the formal notions of:

$$Q = (Oid, L, C),$$

with *Oid* the node, C the conditions of the instances in *L*. The equivalent query via the *Cypher* query language would be:

MATCH (*L:Node{name:'Final-node'}*)
WHERE *tweet.likes > 300 and tweet.mentions >5*
RETURN (*Oid*)

So, to retrieve meanly information, we have to select the node or the interaction that are interested to us. We have tried different queries using *Cypher* with very interesting results. As an example, we show network and the node retrieved using this Query (Figure 11).

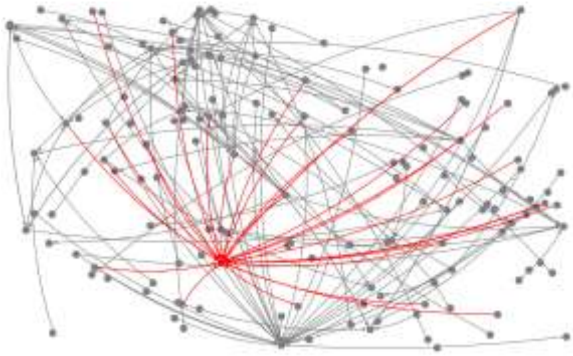


Fig. 11. Automatic selection using *Cypher* language.

6 Conclusions

Complex graphs, contains thousands of nodes of high degree, that are difficult to visualize. Displaying all of the nodes and edges of these graphs can create an incomprehensible cluttered output. We have presented a simplification algorithm that may be applied to a complex graph issue of a Social Media Network, in our case a Twitter network, in order to produce a simplify graph. This simplification was proposed by the use of morphological operators, that are based on Mathematical morphology.

We have represented the Social Media Network as a complete Lattice. In doing this, mathematical morphology has been developed in the context of a relation on a set. It has been shown that this structure is sufficient to define all the basic operations: dilation, erosion, opening and closing, and also to establish their most basic properties.

The simplification of the graph provides an approach to visualizing the fundamental structure of the graph by displaying the most important nodes, where the importance may be based on the topology of the graph and their interaction. The simplification algorithm consists in the iterative use of Opening and Closing operations that cause a growing or shrinking effect in the graph. This process generates the simplification of the network.

As can be seen from this paper, SMA have been and currently are a prominent topic in Network s analysis and simplification. With the advent of the so-called Big Data, we expect this trend to be extremely persistent [29] and promising for opening novel research directions. Indeed, there is no reason to restrict the application of this simplification process the very same ideas we have described here to networks. Any kind of data can be processed with these techniques, notably, image processing.

A special note concerns the relation of community detection to the emergence of core periphery structure in Social Media networks. The process explained in this article, focuses on simplification Network and not in community detection, though we believe that using the correct parameters and exploiting on a correct way the network query language proposed (*Cypher*) some interesting contribution and direction could be envisaged.

In the proposed method based on morphological simplification, we have realized that the parameterization is a fundamental step and we must dedicate special attention to get a homogeneous simplification of nodes and interactions.

This parameterization leads the process of simplification by physical characteristics of the graph and permits to interpret in a simple way the relationship among the nodes, interactions and all characteristics associated.

The notions of dilation, erosion, opening and closing for graph simplification described here appear to be novel, and there is further work needed in order to determine the algebraic properties of these operations. In terms of practical applications, a theory of graphs, perhaps in connection with other forms of granularity for graphs, might be used to model networks at multiple levels of detail.

Future work may be to design query-based simplification techniques that would take user’s interests into account when simplifying a network. It would also be interesting to combine different network abstraction techniques with network simplification, such as a graph compression method to aggregate nodes and interactions. Also, it would be interesting to develop additional importance metrics, as well as testing and evaluating our approach with other simplification methods and on other types of graphs.

References

1. Serra, J., Soille, P.: Mathematical Morphology and its applications to image processing. Kluwer Academic Publishers (1994)
2. Serra, J.: A lattice approach to image segmentation. Journal of Mathematical Imaging and Vision. (24), 83–130 (2006)
3. Bonato, A.: A Course on the Web Graph, American Mathematical Society Graduate Studies Series in Mathematics, Providence, Rhode Island (2008)

4. Facebook statistics, http://www.facebook.com/press/info.php?statistics, last accessed 2017/07/01

5. Twitaholic, http://twitaholic.com/, last accessed 2017/07/01

6. Kleinberg, J.: The small-world phenomenon: An algorithmic perspective. In: Proceedings of the 32nd ACM Symposium on Theory of Computing (2000)

7. Adamic, L. A., Buyukkokten, O.,Adar, E.: A social network caught in the web. First Monday, 8 (2003)

8. Java, A., Song, X., Finin, T., Tseng, B.: Why we twitter: understanding microblogging usage and communities. In: Proceedings of the Joint 9th WEBKDD and 1st SNA-KDD Workshop (2007)

9. Kumar, R., Novak, J., Tomkins, A.: Structure and evolution of on-line social networks. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2006)

10. Newman, M. E. J. Park, J.: Why social networks are different from other types of networks. Phys. Rev. E 68 036122 (2003)

11. Qiu, H., Hancock, E.: Graph simplification and matching using commute times. Pattern Recognition 40, 2874–2889 (2007)

12. Frishman, Y., Tal, A.: Multi-level graph layout on the GPU. Transactions on Visualization and Computer Graphics 13, 6 Nov, 1310–1319 (2007)

13. Rizzi, S.: A genetic approach to hierarchical clustering of Euclidean graphs. Pattern Recognition 2, 1543–1545 (1998)

14. Ban, T., Sen, D.: Graph based topological analysis of tessellated surfaces. In Proc. Eighth ACM Symposium on Solid Modeling and Applications, 274–279 (2003)

15. Kao, M., Occhiogrosso, N., Teng, S.: Simple and efficient graph compression schemes for dense and complement graphs. Journal of Combinatorial Optimization, 2(4), 351–359 (1998)

16. Girvan, M., Newman, M. E. J.: Community Structure in Social and Biological Networks. PNAS 99(12), 7821–7826 (2002)

17. Davood R., Stephen, C.: Effectively Visualizing Large Networks Through Sampling. 16th IEEE Visualization 2005, (VIS 2005), 48 (2005)

18. Gilbert, A., Levchenko, K.: Compressing network graphs. In Proc LinkKDD 04 (2004)

19. Serra, J.: Image Analysis and Mathematical Morphology, Academic Press, London (1982)

20. Heijmans, H.: Morphological Image Operators. Advances in Electronics and Electron Physics Series, Academic Press, Boston (1994)

21. Najman, L., Talbot, H.: Mathematical Morphology: from Theory to Applications. ISTE-Wiley (2010)

22. Vincent, L.: Graphs and mathematical morphology. Signal Processing 16, 365–88 (1989)

23. Flouzat, G., Amram, O. : Segmentation d'images satellitaires par analyse morphologique spatiale et spectrale. Acta Stereologica, 16, 267–274 (1997)

24. Zanoguera, F.: Segmentation interactive d'images fixes et séquences séquences vidéo basée sur des hiérarchies de partitions. Thèse de Doctorat en Morphologie Mathématique, ENSMP (2001)
25. Amous, I., Jedidi, J., Sèdes, F.: A contribution to multimedia document modeling and organizing. 8Th International conference on Object Oriented Information Systems, OOIS'02, Springer LNCS. 2425, 434–444 (2002)
26. Chrisment, C., Sedes, F.: Multimedia Mining, A Highway to Intelligent Multimedia Documents. Multimedia Systems and Applications Series, Kluwer Academic Publisher, V. 22, 245 (2002)
27. Holzschuher, F., Peinl, R.: Performance of graph query languages: comparison of cypher, gremlin and native access in Neo4j. In Proceedings of the Joint EDBT/ICDT Workshops. 195–204, ACM (2013)
28. Lafferty, J., Zhai, C.: Document language models, query models, and risk minimization for information retrieval. In ACM SIGIR Forum. 51(2). 251–259 (2017)
29. Zwaenepoel, W.: Really Big Data: Analytics on Graphs with Trillions of Edges. In LIPIcs-Leibniz International Proceedings in Informatics, 70 (2017)

ISSN 1870-4069

WSDL Information Selection for Improving Web Service Classification

Christian Sánchez-Sánchez[1], Esaú Villatoro-Tello[1],
Gabriela Ramírez-de-la-Rosa[1], Héctor Jiménez-Salazar[1], David Pinto[2]

[1] Universidad Autónoma Metropolitana Unidad Cuajimalpa,
Departamento de Tecnologías de la Información,
División de Ciencias de la Comunicación y Diseño, México

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

{csanchez, evillatoro, gramirez, hjimenez}@correo.cua.uam.mx,
dpinto@cs.buap.mx

Abstract. Currently, the increasing number of available Web Services (WS) over the Internet has induced the urgency for proposing new ways for searching and categorizing such software pieces. Normally, WS functionality is detailed through the WSDL description language, resulting in a structured document that includes a great variety of features definition. One of the WSDL inner features *"documentation"* is designed to describe the Web Service functionality, in natural language, which could help to classify and find WS. Nevertheless, the majority of WS lack of that description. To tackle this problem, this paper presents an analysis of the WSDL inner feature information that can assist to classify WS, without any extra data. The experiments carried out on three different WSDL collections showed that only with minimal information is possible to increase the performance of automatic WS classification.

Keywords. Web service classification, WSDL information analysis, feature selection.

1 Introduction

Nowadays, in the field of Software Development exists a strong motivation for preserving and encouraging the use of certain programming styles and conventions as recommended practices. Among the advantages of such conventions are the following: *i)* high efficiency of software applications by means of distributed systems, *ii)* collaborative applications through standard mechanisms, *iii)* loosely coupled systems that allow software reuse, and, *iv)* costs reduction during the software development phase.

Accordingly, Web Services (WS) emerged as software development mechanisms that allow developers to fulfill previously mentioned characteristics. A WS can be thought as a web application which uses XML based standards for

pp. 83–96; rec. 2017-09-05; acc. 2017-10-20

communicating with external systems for providing the necessary service for the user [2]. Web services being a business trend over software applications, contain encapsulated descriptions of their functionality (*i.e.*, methods and functions, usually known as *description features*) described as an abstract interface by means of using standard Web Services Description Language (WSDL). For locating desired services, users appeal for public registries like UDDI (Universal Description, Discovery and Integration) where by means of matching their requirements to a set of registered services, users are able to obtain "relevant" services. However, search functionality still is very simple and fails to account for relationships between WS and users' real needs [3]. A bigger problem is that not all WS developers register their services and with the ever-increasing amount of published WS on the Internet, the task of finding the correct service has become a challenging issue in service-oriented computing [2,3].

Another important aspect to consider is that in spite of fact that the WSDL description is a structured document, it is hard for a common Internet user to understand its content. One of the WSDL inner features *"documentation"* is designed to describe the Web Service functionality in natural language, which could help to classify and find WS. Nevertheless, it is very common that suppliers do not include such *documentation.*

This is why the necessity of creating algorithms or methods that help in the process of categorize and search WS becomes important. For this reason, in order to fulfill that necessity, some questions were stated. *(a)* Is it possible to extract enough understandable information (words) from WSDL documents?, and *(b)* which information, from WSDL inner features, improve WS classification?

In order to answer these questions, this paper presents an analysis of the WSDL inner feature information that can assist to classify WS without any extra data. One characteristic of such information is that is expressed by words.

For the experiments we used WSDL standard collections, namely OWLS-TC[3] v3 and v4, and ASSAM[4]. We show that some of the evaluated features improve the classification performance achieving competitive results to the state of the art.

The rest of this document is organized as follows. Section 2 presents some related work concerning the automatic web services classification task. Section 3 describes how an WSDL document is conformed. Then, Section 4 describes used datasets, experimental setup and obtained results by our proposed approach. Finally, Section 5 depicts our conclusions and some future work directions.

2 Related Work

In the literature of WS classification, two different types of approaches can be distinguished. Those that use external resources to build the classification method, and those that do not need any external resources but just the *description* features contained in the WSDL document.

[3] http://projects.semwebcentral.org/projects/owls-tc/

[4] http://www.andreas-hess.info/projects/annotator/

On the one hand, among the external resources used by some approaches, the most widely used is The United Nations Standard Products and Service Code (UNSPSC), a standard taxonomy used to manually classify WS. This taxonomy defines a five-level and tree-structured hierarchical classification. The UNSPSC taxonomy was used by [9] to automatically classify WS using clustering algorithms in two steps. First, it considers the terms contained in the metadata of the WSDL documents to generate a tree structure representation of the WS, and then it considers the underlying semantic relation among metadata structures, such as, terms co-occurrences of words taken from the input, output and function descriptions of the WSDL document: *its description features*. After these two clustering results, Liang et al. [9] uses the taxonomy to assign a class to a WS tree.

In the approach proposed by Wang et al. [16] the UNSPSC taxonomy is used to generate a set of vectors for the training phase of the SVM algorithm. Given a set of domains, each subtree found in the taxonomy under these domains are treated as concept of that domain. They claim that the functional description of a WS is always related to a set of concepts. Therefore, a vector is constructed for each concept to represent a training document for the SVM algorithm. In addition to UNSPSC taxonomy, in this work they used WordNet to provide semantic similarity of concepts to weight the terms in the vector space model. WordNet as external resource was also used by Boujarwah et al. [4] as a lexical English database to generate conceptual graphs for each domain. Then, they used the conceptual graphs generated to classify a new WS.

Another example of this kind of approaches is introduced by Yang et al. [18]. Using OWLS-TC4 dataset, words are extracted from the WSDL. An external resource is used in order to identify abbreviations or if they are nouns or verbs. The pre-processing step involves: splitting (verbs and nouns), eliminating stop words, stemming and removing specific tags (web, service, input, output). Then they tested 4 different classification algorithms Support Vector Machines (SVM), Naive Bayes (NB), Decision Tree (DT) C4.5 and Neuronal Networks. Only words from names of services, operations, inputs and outputs were extracted. The best results for classification was obtained using output name words and applying C4.5 DT Algorithm.

In the approach introduced by Nisa et al. [11] were extracted: service name, service documentation, WSDL messages, WSDL ports and WSDL schema from WSDL documents, in order to classify web services using text mining. For each extracted feature that was classified, using Maximum Entropy, a comparison of the accuracy, through different categories, was done. The best results were obtained including WSDL Schemas information. In this paper it is also showed a comparison of the effects of using some preprocessing like: steeming, lemmatization and word splitter. The results were improved using the last two. Unfortunately, their dataset is not available for comparison.

On the other hand, there are WS classification methods developed without using any external resource to classify WS, these methods usually rely on the WSLD *description* features only (Section 3). For instance, Saha et al. [12]

propose a WS representation based on Tensor Space Model (TSM) in order to capture the internal structure of WSDL documents. The method consists in selecting a set of relevant tags from a WSDL document. For each tag, they build a tensor using all words under that particular tag. For each tensor they apply a classification algorithm that gave independent classification results and as a final step they combine all this information using rough sets. Another example is given by Bruno et al. [5], where for classifying a WS, authors identified key concepts in the WSDL *description features*, and then by means of a SVM algorithm, they classify each WS into a specific domain.

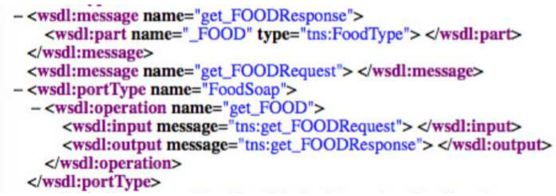
Notice that all related work shown above used, in some way, the information of the WSDL description file. However, some works use it merely as source of terms to construct more elaborate representation, and some of them use it as the main source of information to the classification. Consequently, these previous research demonstrate, to some extent, that there exists a relationship between WS functionality and the information contained in the WSDL document, opposite to what Lu *et al.* claim [10].

Contrary to previous work, our proposed approach do not depend on any external resources during the classification phase. Also, the information gotten from the WSDL inner features are words, many of previous approaches worked using strings.

3 A WSDL Document

As we have mentioned in previous sections, employing the WSDL language for describing WS functionality is a recommended practice among software developers. Accordingly, as examples of the information that a user can obtain from reviewing the WSDL document (*i.e., the description features*) we have the following:

- **Types-** a container for data type definitions using some type system (*i.e., data types*).
- **Messages-** an abstract typed definition of the data being communicated, messages normally tend to include parameters information and communication protocols.
- **Operations-** an abstract description of all actions supported by the web service (*i.e.,* methods' *names*).
- **Documentation-** natural language description of the full functionality (operations) of the web service (usually missing).
- **Port Type-** an abstract set of operations supported by one or more endpoints.
- **Binding-** a concrete protocol and data format specification for a particular port type.
- **Port-** a single endpoint defined as a combination of a binding and a network address.
- **Service-** a collection of related endpoints.



```
– <wsdl:message name="get_FOODResponse">
    <wsdl:part name="_FOOD" type="tns:FoodType"> </wsdl:part>
  </wsdl:message>
  <wsdl:message name="get_FOODRequest"> </wsdl:message>
– <wsdl:portType name="FoodSoap">
  – <wsdl:operation name="get_FOOD">
      <wsdl:input message="tns:get_FOODRequest"> </wsdl:input>
      <wsdl:output message="tns:get_FOODResponse"> </wsdl:output>
    </wsdl:operation>
  </wsdl:portType>
```

Fig. 1. An excerpt of a real WSDL document from the domain FOOD extracted from the OWLS-TC V3 collection. Notice that any *documentation* has been provided.

According to Stroulia et al. [15], information like services' names associated to its methods, parameters and data types are useful since they reflect (to some extent) the semantics of the underlying capabilities. For this reason in the proposed experiments, it is contemplated extracting information from inner features and test which of them are useful to improve classification. Next, more about the information extraction algorithm and document representation is given.

3.1 Information Extraction Algorithm

For performed experiments we considered the following features: *Service Name*, *Operations*, *Documentation*, *Messages*, and *Types*, configured in different ways. Observe that only the *Documentation* feature represents a text described in natural language, hence, extracting information of such feature do not represent a complicate process. However, for the rest of the considered features, extracted terms do not represent well formed words (See Figure 1), therefore we followed the steps described in Algorithm 1 in order to extract more readable information (words).

Notice that the proposed algorithm uses a method named **generateSubstrings**, which aims at extracting all possible sub-strings from any composed expression e_i extracted from the *description features*. Proposed method performs the following steps:

- Obtains all possible sub-strings from expression e_i by means of computing all the combinations of consecutive characters from e_i.
- Obtains all sub-strings that start with a capital letter. Intuitively, composed expressions will contain in fact several words, and as a common programming practice, developers tend to set the first letter of each word in its capital form.
- Obtains all possible sub-strings from e_i separated by some special character. Similarly to previous point, it is also a common practice among developers to use some set of special characters to separate composed expressions (*e.g.*, .− =:,).

Finally, as can be noticed in Algorithm 1, once all possible sub-strings were obtained, each word is checked for its existence within WordNet. We verify if that word is not a stop word[5]. Thus, at the end we have a set of words, extracted from the *description features*. And those extracted words are used for classifying the WS. It is worth mentioning that our proposed method employs WordNet for identifying readable words only, but not for the classification process nor to include other terms, which has been the traditional approach from previous works (see section 2).

```
input : expression $e_i$ extracted from the description
        features and a list of stop words $swList$
output: list ($listOfWords$) of readable words extracted
        from $e_i$

//If the given expression $e_i$ is a known word in WordNet (WN)
  then $listOfWords$ is composed by $e_i$
if (($e_i \in WN$) and ($e_i \notin swList$)) then
    Add $e_i$ to $listOfWords$;
    return $listOfWords$;
end
//If FALSE, we begin the process to extract all possible
  words from $e_i$
else
    $substrings \leftarrow$ generateSubstrings($e_i$);
    //For each obtained sub-string from $e_i$ we verify if it
      represents a readable word
    for $i \leftarrow 1$ to $|substrings|$ do
        if (($substrings[i] \in WN$) and ($substrings[i] \notin swList$)) then
            Add $substrings[i]$ to $listOfWords$;
        end
    end
    return $listOfWords$;
end
Exit;
```

Algorithm 1: Proposed algorithm for extracting information from WSDL features.

3.2 Document Representation

As we have mentioned before, we face the problem of Web Services classification as a *document classification* task. Although document classification subsumes two types of text analyses: clustering and categorization. The difference between

[5] Normally, stop words are formed by short function words, such as *the, is, at, which*, and *on*; *i.e.,* prepositions, pronouns, etc.

the two is that the latter uses a predefined number of classes or categories with their corresponding tags, whereas in the former approach, the number and the tag for each category is to be discovered. Since in categorization the classes are known a priori, categorization algorithms usually take advantage of them by using supervised algorithms with some kind of training step.

Accordingly, Text Categorization (TC) is the task of automatically sorting a set of documents into categories (or classes, or topics) from a predefined set [13]. In its simplest form, the text classification problem can be formulated as follows: Given a set of training documents $\mathcal{D}_{Tr} = \{d_1, \ldots, d_n\}$ and a set of predefined categories $\mathcal{C} = \{c_1, \ldots, c_m\}$, the goal of TC is to devise a learning algorithm that is able to generate a classification model (*i.e.,* hypothesis) $h : \mathcal{D} \to \mathcal{C}$ that will be able to accurately classify unseen documents from $\mathcal{D}$.

The design of learning algorithms for text categorization has usually followed the classical approach of the pattern recognition field, where data instances (*i.e.,* documents) first undergo an appropriate representation. Accordingly, the first step corresponds to the *indexing* of training documents ($\mathcal{D}_{Tr}$), where each document d_j is transformed into a compact form of its content. Commonly, each document is represented as a vector of weighted terms; such idea is taken from the Vector Space Model proposed in the field of Information Retrieval [1]. Therefore, given a document $d_j \in \mathcal{D}_{Tr}$, it is represented as a vector $\overrightarrow{d}_j = \langle w_{kj}, \ldots, w_{|\tau|j} \rangle$, where τ depicts the *dictionary*, *i.e.,* the set of different terms (words) that appear at least once in some document of $\mathcal{D}_{Tr}$, and w_{kj} establishes the importance of term t_k within document d_j.

Normally, τ is obtained from filtering words from the document collection, in other words, τ is the result of a pre-processing step. As pre-processing step all stop-words were removed. Once τ has been defined, in order to represent the documents $d_j \in \mathcal{D}_{Tr}$, we employed the well known Bag of Words (BOW) paradigm.

The BOW representation has been the traditional form for representing documents [13]. Such approach employs single words as elements of the vector of terms. There are several proposals for computing the weight $w_{k,j}$ of each term (*i.e.,* the importance of each term/word). Among the most successful weighting strategies are: the boolean weight, term frequency and relative term frequency. Next we briefly describe each one of these weighting schemes.

- *Boolean weighting:* It assigns a weight of 1 if the term t_k appears within the document d_j, otherwise the value assigned is zero:

$$w_{kj} = \begin{cases} 1, \text{if } t_k \in d_j \\ 0, \text{otherwise.} \end{cases} \tag{1}$$

- *Term frequency weighting (TF):* For this particular case, the assigned weight will be equal to the number of times the term t_k occurs within document d_j:

$$w_{kj} = f_{kj}. \tag{2}$$

- *Relative Term Frequency weighting (TF-IDF):* This type of weighting scheme represents a variation from the TF technique. For computing the TF-IDF weight we must follow:

$$w_{kj} = TF(t_k) \times IDF(t_k). \tag{3}$$

where $TF(t_k) = f_{kj}$, in other words, it represents the frequency value of term t_k within document d_j. IDF is also known as the "inverse frequency" of term t_k within document d_j. The IDF value represent to some extent how "rare" is term t_k. For computing the IDF value we follow:

$$IDF(t_k) = \log \frac{|D|}{\{d_j \in D : t_k \in d_j\}}. \tag{4}$$

where D represents the document collection that is being indexed, *i.e.*, $\mathcal{D}_{Tr}$

3.3 Classifiers

Since our proposal for WSDL document representation does not dependent of a particular learning algorithm we can use any classifier to face the WS classification problem. For our experiments we selected 3 different learning algorithms which are representative of the wide diversity of methods available in the machine learning field [6, 8]. Specifically, we considered the following classifiers:

- **Naïve Bayes (NB).** A probabilistic-based method that assumes attributes are independent among them given the class.
- **Support vector machine (SVM).** A linear discriminant that aims to find an optimal separating hyperplane; a linear kernel was used for this work.
- **J48.** An algorithm used to generate a decision tree, which select the most discriminating features based on its entropy measure.

We employed the Weka implementation of the above described algorithms; default parameters were considered for all the performed experiments [7].

4 Experimental Setup

4.1 Datasets

For all the experiments performed we used a subset of the ASSAM web services collection. This collection is made up by real Web Services description documents, obtained from Salcentral [14] and Xmethods [17]. The WSDL documents are organized into a class hierarchy, that in some cases have sub-classes with at most two levels depth. Originally, this collection had 814 WSDL documents distributed in 26 classes, however, in order to prove the pertinence of our proposed approach we modified the collection applying the following steps: *i)* we flatten all classes, *i.e.,* we suit WSDL documents contained in parent classes in some particular sub-classes, and *ii)* we discarded all WSDL documents that do not contain at least one word (recognized by WordNet) for each considered *description* feature (see Table 1 for details). Due to many of WS Descriptions have the *documentation* feature, the classification of the information extracted from this feature was taken as a baseline.

In addition to the previous collection, we also performed experiments using the OWLS-TC collection (version 3 and version 4). Together the two datasets contain more than 1000 WS, covering seven and nine categories respectively. Contrary to the ASSAM collection, the documents of the collections OWLS-TC V3 and V4 do not contain the *documentation* feature. Table 1 shows some statistics from the considered datasets.

Table 1. Basic statistics from the considered datasets.

	ASSAM	**TC-V3**	**TC-V4**
Num. docs	203	1006	1082
Vocabulary	2829	283	378
Num. classes	22	7	9
Docs by class (Avg.)	9.2	143.7	120.2
Documentation (DOC)	Yes	No	No
Terms in DOC (Avg.)	26.3	0	0

Notice that although the TC V3 and V4 are larger collections (*i.e.,* they have more WSDL documents), their vocabulary is smaller than the ASSAM collection. The reason for this difference is that, documents from the ASSAM collection do have the *documentation* feature. This characteristic can be observed also in the last row of Table 1 (*i.e., Terms in Doc*), which indicates the average number of terms for each WSDL document contained in the *documentation* feature.

It is also important to mention that the ASSAM collection represents a more complicated challenge for classifications systems, just by the fact that it contains more categories (22), and as a consequence, less examples for each class are available (9.2 in average).

4.2 Evaluation

For evaluating the classifiers performance we adopted standard measures from the text-categorization field. The leading evaluation measure is the macro F_1 measure, defined as follows:

$$Macro - F_1 = \frac{1}{|K|} \sum_{C_i \in C} \left[\frac{2 \times R(C_i) \times P(C_i)}{R(C_i) + P(C_i)} \right].$$

where the per-class recall (R) and precision (P) measures are defined as follows:

$$R(C_i) = \frac{number\, of\, correct\, predictions\, of\, C_i}{number\, of\, examples\, of\, C_i},$$

and

$$P(C_i) = \frac{number\, of\, correct\, predictions\, of\, C_i}{number\, of\, predictions\, as\, C_i}.$$

It is worth mentioning that during our experiments we applied a 10 cross-fold validation strategy.

4.3 Experimental Settings

The main goal of our experiments was to evaluate which of the information extracted from WSDL inner features improves WS classification.

Accordingly, we considered for our experiments the following *description features*: messages (*Msgs*), operations (*Names*) and types (*Params*)[6].

Hypothetically, the *Names* attribute (*i.e.,* name of a method in the WS) might be part of the selected information, which it could be more precise when includes the *Parameter* attribute (*i.e.,* names of the parameters or data types). Furthermore, if we combine these *description features* (*Names* and *Parameters*) with the *Messages* attribute we expect that it will also contain enough information to allow an automatic classifier to correctly define its category. Consequently, we defined our experiments by means of using single features and its respective combinations.

4.4 Results

The obtained experimental results are reported in Tables 2, 3 and 4 in terms of macro F_1. Results marked in **bold** indicate the *best* results obtained over different configurations.

Table 2 shows the results obtained when the ASSAM WSDL document collection is used. The first column (*i.e.*, "Description features") indicates the *description features* from which information was extracted. Notice that better results are obtained when a *boolean* representation is employed, which means that just by the presence of certain words it is possible to assign the WS category.

Table 2. Results of the experiments performed using the ASSAM collection.

Description	BOOLEAN			TF-IDF		
features	**NB**	**SVM**	**J48**	**NB**	**SVM**	**J48**
Msgs	0.37	0.38	0.34	0.27	0.31	0.31
Names	0.41	**0.43**	0.34	0.32	0.35	0.32
Params	0.39	0.39	0.34	0.33	0.36	**0.36**
Names+Msgs	0.41	0.41	**0.36**	0.25	0.34	0.35
Names+Param	0.42	**0.43**	0.34	**0.36**	**0.38**	0.27
Msgs+Params	0.40	0.35	0.32	0.33	0.36	0.30
Names+Msgs+Params	**0.44**	0.40	0.32	0.34	0.37	0.29

The obtained results indicate that when the *Names* feature is involved in the information extracted better results are reached. Although the best result was

[6] Refer to Section 3 to view a detailed description of the selected features.

obtained by the combination of *Names+Msgs+Params* employing a Bayesian classifier, but the combination of *Names+Params* allows a more stable behavior. This situation indicates (to some extent) that methods' names as well as the parameters have an important role on the definition of a Web service functionality, hence providing important elements to define the WS category.

Table 3 and Table 4 exhibit the results obtained on the OWLS-TC V3 and V4 collections respectively. Similarly to the results obtained on the ASSAM collection, better results are obtained under the combinations of the *Names* and *Params* description features.

Table 3. Results of the experiments performed using the OWLS-TC V3 collection.

Description	BOOLEAN			TF-IDF		
features	**NB**	**SVM**	**J48**	**NB**	**SVM**	**J48**
Msgs	0.48	0.55	0.37	0.47	0.55	0.37
Names	0.71	0.77	0.68	0.70	0.79	0.70
Params	0.73	0.78	0.79	0.62	0.80	0.80
Names+Msgs	0.71	0.77	0.68	0.69	0.78	0.70
Names+Param	0.79	**0.86**	0.82	**0.77**	0.85	0.83
Msgs+Params	0.77	0.85	**0.83**	0.73	0.85	**0.85**
Names+Msgs+Params	**0.80**	**0.86**	0.82	**0.77**	**0.86**	0.83

Notice that a better performance is obtained under a *boolean* weighting scheme. This indicates that WS categories can be determined by just the presence of certain words, which lead us to consider that counting the frequencies downgrades the classifier's performance.

Table 4. Results of the experiments performed using the OWLS-TC V4 collection.

Description	BOOLEAN			TF-IDF		
features	**NB**	**SVM**	**J48**	**NB**	**SVM**	**J48**
Msgs	0.47	0.52	0.37	0.44	0.53	0.37
Names	0.70	0.76	0.67	0.69	0.78	0.68
Params	0.74	0.79	0.78	0.65	0.80	0.81
Names+Msgs	0.70	0.76	0.67	0.69	0.75	0.68
Names+Param	**0.81**	**0.86**	**0.82**	**0.77**	**0.86**	0.84
Msgs+Params	0.78	0.85	**0.82**	0.75	**0.86**	**0.85**
Names+Msgs+Params	**0.81**	**0.86**	**0.82**	**0.77**	**0.86**	0.84

Similarly to the results obtained when the ASSAM collection is used, the combination of *Names+Params* allows to the classifier to reach an acceptable classification performance on both, the OWLS-TC V3 and V4 datasets. These particular results reinforce our intuition that extracting only information from some WSDL features can improve classification.

Finally, it is important to mention that the results obtained on the OWLS-TC V3 (Table 3) are directly comparable against results reported on [16] and [4]. On the one hand, Wang et al. [16] reports an F measure of 89% using a SVM with a lineal kernel. However, it important to remember that their proposed approach depends on the UNSPSC taxonomy (see Section 2).

On the other hand, the work proposed in [4] reports an average precision (P) of 65% and an average recall (R) of 70%. Their proposed approach, similarly to [16], depends on the UNSPSC taxonomy and WordNet (see Section 2). Although our results are reported in terms of the macro F_1, the average precision and average recall for our best configuration (*i.e., Names+Msgs+Params* using a SVM classifier in Table 3) are 87% and 86% respectively. In conclusion, our proposed approach is able to obtain a competitive performance against the state-of-the-art methods [4, 16] without needing or employing any external resource during the classification stage.

4.5 Additional Experiment

As we have mentioned in previous sections, it is believed that considering only the information contained in the *documentation* feature can improve WS classification. Accordingly, an intuitive comparison of such a system would be to contrast the obtained classification results against a system that uses the original *documentation feature* for representing and classifying WSDL documents.

On section 4.1 we showed that the only dataset that actually contains the *documentation* feature is the ASSAM collection (See Table 1). Hence, we performed an additional experiment over the ASSAM collection, where our main goal was to explore the pertinence of this particular feature when classifying WSDL documents (See Figure 2).

Notice that by using only the original *documentation feature* (*Doc*) the classification performance gets the worse results. Particularly for the Bayesian classifier, goes from a macro F_1 of 44% using the *Names+Msgs+Params* features to a 37% using only the original *documentation* feature. This result indicates, to some extent, that documentation provided by WS suppliers tends to be ambiguous and it introduces noisy elements to an automatic classification system.

Finally, it is worth mentioning that according to the *paired Student's t-test* - with a confidence level of 99 percent - the improvement obtained using *Names + Params* with both classifiers (*i.e.,* NB and SVM) are statistically significant over the results obtained employing only the *documentation feature.*

5 Conclusions

We have introduced an analysis of the combination of WSDL inner feature information that can assist to classify WS, without any extra data. These in order to select the information that improves WS categorization.

We report experiments on three different WSDL collections; the obtained results indicate that selecting only information from some WSDL inner features

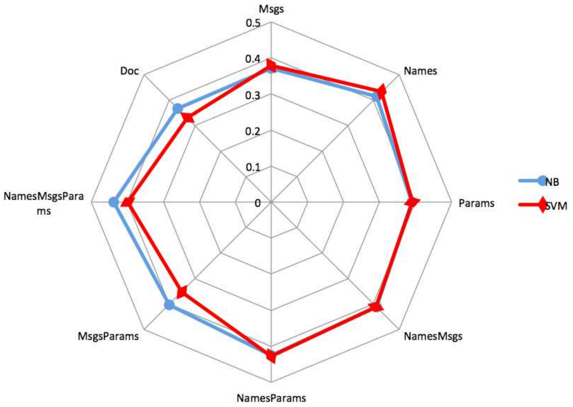


Fig. 2. Comparison of the classification performance when the *documentation* feature is employed to represent WSDL documents against using several *description* features for the construction of the proposed *virtual* feature. All the experiments were performed using the ASSAM collection.

can obtain a competitive performance against the state-of-the-art methods. The performed experiments showed that by means of combining the operations' names and parameters it is possible to obtain a macro F_1 of 86% for the OWLS-TC collections. Similar results were obtained on the ASSAM collection when we used the same combination of information extracted from WSDL features.

An additional experiment showed that the names and parameters information combination allows better classification results compared to those obtained when WSDL documents are represented by means of the *documentation* feature. A more deeper analysis demonstrate that the information from selected *description features* are able to generate a less complex and more general description of WS functionality. On the contrary, by using only the *documentation* feature results in a more complex and highly specific description of WS functionality, which leads to a less accurate classification process.

Future work directions include using Distributional Term Representations which have proven to be effective on reducing the effect of low term frequency occurrence, sparsity and term ambiguity, which are common characteristics of WSDL documents.

Acknowledgments. This work was partially funded by CONACYT under the Thematic Networks program (Language Technologies Thematic Network project 281795).

References

1. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. Addison Wesley (1999)

2. Balasubramanian, D.L., Murugaiyan, S.R., Sambasivam, G., T., V., Dhavachelvan, P.: Semantic web service clustering using concept lattice: Multi agent based approach. International Journal of Engineering and Technology 5(5), 3699–3714 (2013)
3. Batra, S., Bawa, S.: Web service categorization using normalized similarity score. Internaitonal Journal of Computer Theory and Engineering 2(1), 139–141 (2010)
4. Boujarwah, E., Yahyaoui, H., Almulla, M.: A new unsupervised web services classification based on conceptual graphs. In: ICIW 2013, The Eighth International Conference on Internet and Web Applications and Services. pp. 90–94 (2013)
5. Bruno, M., Canfora, G., Penta, M.D., Scognamiglio, R.: An approach to support web service classification and annotation. In: EEE. pp. 138–143. IEEE Computer Society
6. Duda, R., Hart, P.: Pattern classification and scene analysis. Wiley (1996), http://www.ica.luz.ve/ enava/redesn/ebooks/DHS/Versi%F3n PS/DHSChap4.ps
7. Garner, S.R.: Weka: The waikato environment for knowledge analysis. In: Proc. of the New Zealand Computer Science Research Students Conference. pp. 57–64 (1995)
8. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer (2009)
9. Liang, Q., Li, P., Hung, P., Wu, X.: Clustering web services for automatic categorization. In: Services Computing, 2009. SCC '09. IEEE International Conference on. pp. 380–387 (Sept 2009)
10. Lu, G., Wang, T., Zhang, G., Li, S.: Semantic web services discovery based on domain ontology. In: World Automation Congress (WAC), 2012. pp. 1–4 (2012)
11. Nisa, R., Qamar, U.: A text mining based approach for web service classification. Information Systems and e-Business Management 13(4), 751–768 (2015)
12. Saha, S., Murthy, C.A., Pal, S.K.: Classification of web services using tensor space model and rough ensemble classifier. In: An, A., Matwin, S., Ras, Z.W., Slezak, D. (eds.) ISMIS. Lecture Notes in Computer Science, vol. 4994, pp. 508–513. Springer (2008)
13. Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys 34(1), 1–47 (2002)
14. Seekda: *http://webservices.seekda.com/* (2012), last visited on September 2014
15. Stroulia, E., Wang, Y.: Structural and semantic matching for assessing web-service similarity. International Journal of Cooperative Information Systems 5(14), 407–437 (2005)
16. Wang, H., Shi, Y., Zhou, X., Zhou, Q., Shao, S., Bouguettaya, A.: Web service classification using support vector machine. In: 22nd IEEE International Conference on Tools with Artificial Intelligence (ICTAI). vol. 1, pp. 3–6. IEEE Computer Society (2010)
17. Xmethods: *http://xmethods.net/ve2/index.po* (2013), last visited on September 2014
18. Yang, J., Zhou, X.: Semi-automatic web service classification using machine learning. International Journal of u-and e-Service, Science and Technology 8(4), 339–348 (2015)

ISSN 1870-4069

A Study on Significance on Features in Emotion Recognition System for Poems

Sreeja Ponnarassery Sreenivasan, G S Mahalakshmi

Anna University, Department of Computer Science and Engineering,
Chennai, India

srj_ps@yahoo.com, gsmaha@annauni.edu

Abstract. Poem is a type of literature designed to express, concepts, emotions, and experiences in an excellent way. The main aim of this work is to recognize emotion automatically, with an emphasis on exploring features of poems composed in English. This is an innovative approach to emotion recognition from poems. A Poem Emotion Recognition System (PERS) is developed to identify emotions from the poems, classified into nine emotions, based on *Navarasa* under *Rasa Theory* which is described in *Natyashastra* written by *Bharatha Muni*. The nine basic emotions such as *Love, Sad, Anger, Hate, Fear, Surprise, Courage, Joy, and Peace classified as Navarasa.* The poems are mined from the web and extracted ten features which will help to identify the emotion depicted by poems. The main contribution of this paper is the feature engineering. The evaluation contains measuring the performance of different feature sets across Naive Bayes classification. This experiment explains grouping of similar features gives different results, and it shows the combination of all feature gives a better result. Similarly, logistic regression identified the significant features in each emotion category.

Keywords. Poem emotion recognition corpus, emotion analysis, Naive Bayes classifier, logistic regression, maximum likelihood probability.

1 Introduction

A surge in rapidly increasing subjective expressions in text media have triggered keen interest in methods that automatically identify opinions, emotions, and sentiments in text. This proposed work explores methods for automatic emotion recognition from poetry written in English. In this research work, a Poem Emotion Recognition System (PERS) has been developed to explore the possibilities and limitations of automatic emotion recognition. Emotions are classified, based on the 'Navarasa' described in the 'Natyashastra' [17]. Navarasa comprises nine basic emotions namely love, sadness, anger, hate, fear, surprise, courage, joy, and peace.

Navarasa is based on Rasa Theory given by Bharatha Muni in Natyashastra. NatyaShastra is an Indian text dated between 2nd century BC and 2nd century AD that analyzes all features of performing art [8]. The main contribution of the research work is as follows:

1. Extraction of Linguistic, Orthographic, Statistical, Semantic and Poetic Features.
2. Development of a poem emotion recognition model using Maximum Posterior Probability and identification of significant features.
3. Identification of most contributing feature in emotion recognition by Logistic Regression.

The proposed PERS approached the problem in two ways, utilizing machine learning methods: developing an emotion model based on the Maximum Posterior Probability and Logistic Regression Method. The rest of this work is as follows, Section 2 provides an overview of related work, Section 3 presents the proposed methodology, and Section 4 gives the details of evaluation results. Finally, conclusions are presented in Section 5.

2 Literature Survey

Classification is a data analysis task in which the model or classifier predicts the category label. A text underscores the emotional state of the writer and evokes emotions in the reader. The emotion of the text can be interpreted in different ways using assorted computational models. As Literary Arts comprises many emotions, these literature pieces especially poems can be used for the task of Emotion Recognition, which is very challenging in computational point of view. Over the past semi-century, there have been multiple approaches to emotion recognition from text, and many emotion recognition types of research based on probabilistic approaches.

Alm et al. [1] used a simple Natural Language Parser for keyword spotting, phrase length measurement, and emotion identification. Wu et al. [26] used semantic labels and a Separable Mixture Model to identify emotions. They manually generated rules for emotion, semantic labels, and attitudes with the help of emotion-generation rules, semantic labels and attitudes. Emotion association rules are automatically derived using the Apriori algorithm.

Strapparava and Mihalcea [23] constructed a large dataset of news headlines annotated for six basic emotions [6] such as anger, disgust, fear, joy, sadness, and surprise. They proposed the Latent Semantic Analysis (LSA) and the Naïve Bayes classifier and evaluated several knowledge-based methods for the automatic identification of these emotions in text. Minato et al. [15] constructed a Japanese Emotion Corpus and identified emotions automatically through an analysis of the corpus. The advantage is that it can yield high precision. However, its disadvantage is that it is impossible to determine the emotion of words that are not in the corpus. Das and Bandyopadhyay [3] used the conditional random field classifier to recognize emotion in sentences in Bengali blogs.

Perfors et al. [16] discussed, in detail, issues and applications of the Bayesian approach in cognitive science. Bielza et al. [2] proposed multi-dimensional Bayesian network classifiers which are probabilistic graphical models that systematize class and feature variables as a class subgraph, a feature subgraph and a bridge (from class to feature) subgraph. Yoon and Chung [27] proposed a classifier based on the Nave Bayes theorem and defined emotions in in a two-level and three-level class. They represented emotions on the arousal and valence dimensions. Lei et al. [11] described how a context-aware system could be easily constructed in different domains by mashing up Bayesian network fractions independently designed or learned. Lee et al. [10] extracted context information using the Bayesian network.

Steyerberg et al. [22] used logistic regression for clinical decision making that requires estimates of the likelihood of a dichotomous outcome in individual patients. In this study, they compared alternative strategies in 23 small subsamples from a large dataset of patients with an acute myocardial infarction and developed predictive models for 30-day mortality. D'Mello et al. [4] explored the reliability of detecting a learner's affect from conversational features extracted from interactions with the Auto Tutor, an Intelligent Tutoring System that helps students learn by holding a conversation in natural language. They used multiple regression analysis and confirmed the hypothesis that dialogue features could expressively predict the affective states of confusion, boredom, frustration, and flow.

3 Poem Emotion Recognition System

This section discusses the proposed novel Poem Emotion Recognition System (PERS). In this paper PERS consists of two experiments. One is the machine learning classification model, Naive Bayes Classifier and a prediction model, Logistic Regression are detailed. Identifying emotions from poems is a classification issue, from the viewpoint of text mining. An interesting task that offers diverse challenges for classification such as the following:

3.1 Poem Emotion Recognition Corpus

Despite the availability of several lexicons in emotion analysis, those that help to identify emotions from poetry are few and far between. The lexicons that are available are not specifically poetry-centric and consequently fail to focus on poetic features. Emotion classification is based on the Navarasa described in the Natyasastra [17]. Navarasa, to recapitulate, consists of nine primary emotions: love, sadness, anger, hate, fear, surprise, courage, joy, and peace. Although there are many text corpora for emotion recognition, we are unaware of the existence of a text corpus for poetry, based on the said nine emotions.

The corpus PERC[1] created is from an exhaustive collection of poetry especially that of Indian poets during the period 1850-2016 and this corpus is publically available now. The novelty of this research is the creation of a corpus using poems mined from the web[2] and evaluated by experts in the field.The corpus has a data size

[1] http://dx.doi.org/10.17632/n9vbc8g9cx.1

[2] http://100-poems.com, http://www.poetry-chaikhana.com, http://www.poetry-chaikhana.com, http://allpoetry.com

of 736 poems by ten leading poets from the period 1850 to the present day. The average number of words per poem, across the eight poets, ranges from 74-284. Table 1 shows the details of the corpus, such as the poet's name and the number of poems collected for each of the poets. Table 2 depicts the number of poems available in each emotion category.

Definition 1:

We define the PERC as a set of P poems of pairs (p, E) where p is a poem and E is the emotion from a set of features. Our collection is drawn from the work of ten leading poets of the period, including Rabindranath Tagore, Sarojini Naidu, Aurobindo, Ananda Murthy, Darshan Singh, Jiddu Krishnamurthi, Lalan, Nazrul Islam, Kamala Das, and Meena Kandasamy. The poems have been amassed from several well-known sites as the first step towards their collection and selection.

Inter-rater reliability [7] is one of the most efficient methods for the evaluation of the corpus. Since the corpus has been assembled by human experts rather than a machine learning technique, we calculate inter-rater reliability using the Fleiss kappa measurement primarily to study the closeness between the annotations. Since our corpus has acquired an inter-rater agreement value of 0.4816, we have proved that the corpus created shows moderate agreement [14], which is reasonably good.

The corpus has a data size of 736 poems by ten leading poets from the period 1850 to the present day. Table 1 shows the details of the corpus, such as the poet’s name and the number of poems collected for each of the poets. Table 2 details the number of poems available in PERC for each emotion category.

Table 1. Corpus Details.

Poet	No. of Poems	Poet	No. of Poems
Rabindranath Tagore	284	Darshan Singh	20
Aurobindo	110	Anandamurthy	16
Sarojini Naidu	90	Lalan	20
Jiddu Krishnamurthy	20	Kamala Das	69
Nazrul Islam	32	Meena Kandasamy	75
Total 736			

Table 2. Poem Details.

Emotion	No. of Poems	Emotion	No. of Poems
Anger	52	Love	154
Courage	64	Peace	82
Fear	54	Sadness	126
Hate	54	Surprise	50
Joy	100	Total	736

3.2 Feature Extraction

Each instance that provides inputs to PERS is characterized by its values on a fixed, predefined set of features or attributes. To aid proficiency, Features clustered into five groups: linguistic, statistical, semantic, orthographic and poetic.

Linguistic Features: Linguistic features include nouns, verbs, adjectives, and adverbs. Words in grammatical classes show significant results in emotion recognition. Stanford Tagger [25] is used for POS tagging, and extracted noun, verb, adjective and adverb classes. These features comprise a dichotomy that is Boolean.

Statistical Features: This feature set includes term frequency and inverse document frequency, which are the most common features used in classification problems. Term Frequency (TF) and Inverse Document Frequency (IDF) are the two statistical features used in this Emotion Recognition model [19]. Generally, TF of a term can be defined as number of times a term occurs in a document and Document Frequency (DF), is defined as no. of documents having that term. Inverse Document Frequency (IDF) is calculated [19] by Equation (1):

$$IDF = log\frac{Total\ No\ of\ Documents}{DF}. \quad (1)$$

Orthographic Features. Orthographic features include negation words, the title of the poem, the last line of the poem, and the repeated lines/refrain. The negation words saved in a list are extracted through mapping. These negation words act as emotion modifiers that play an important role in emotion identification. The title of the poem is an essential feature in Emotion Recognition (ER) as it offers clues about the content of the poem. The last line irrefutably concludes the poem. This line is, most of the time, the one that carries the most emotion in the poem. The repeated lines that constitute the refrain are another strategic feature, with the repetition underscoring their significance in the poem.

Poetic Features. In this set, only two features similes and metaphors are targeted. The rules that extract these two features are explained below.

Rule 1: If the word$_i$ is like, and the POS tag of like is not equal to the verb, the word$_{i-1}$ and the word$_{i+1}$ are compared, and a simile is found to exist.

Rule 2: If the word$_i$ is as the word$_{i-1}$ and the word$_{i+1}$ two are compared, a simile is found to exist.

Rule 3: If the word$_i$ is are, is, was and were, and the word$_{i+1}$ is not in the present continuous tense form of the verb, a metaphor is found to exist.

Example 1:

"Let thy hue-winged lyrics hover like birds
Over the swirl of the heart's sea. "
- Musa Spiritus, Aurobindo.[3]

In this example, a simile is used when lyrics are compared to birds.

Example 2:

I had gone a-begging from the door in the village path, When thy golden
Chariot appeared in the distance like a gorgeous dream, And I wondered who

[3] https://allpoetry.com/Musa-Spiritus

Was this king-of-all kings!"
- I Had Gone A-begging, Rabindranath Tagore[4]

In this example, the golden chariot is a metaphor for riches and wealth. The chariot, Gilded in gold, indicates that the man to whom it belonged was immensely wealthy.

Semantic Features. Approaches to identifying emotions have evolved from keyword-based methods to semantic-based methods. Semantic-based approaches use semantic-based dictionaries or a semantic knowledge base. They help to identify the conceptual information associated with emotions. A conceptual model is illustrated for concept identification from poems that principally relies on the conceptual information extracted from ConceptNet [13].

ConceptNet [13] is a semantic network of commonsense knowledge made up of over a million simple sentences contributed by volunteers on the Open Mind Common Sense website [12]. This problem is approached by adding a weight that defines a function *C(S, wt)* to identify the concept of a given poem [21]. *S* is a set of relations *S= a, r, c* where *a* is a word in the poem, *r* the relation, and *c* the concept. *wt* is the weight assigned to each relation in ConceptNet. The concept can be defined as follows.

Definition 2:

Concept is defined as a key value pair, *c = (cid; wt),* where *cid* refers to the unique id of the concept and *wt* is the weight assigned to the respective concept [21].

3.3 Poem Emotion Recognition Using Naive Bayes Classifier

The classification technique in Naive Bayes Classifier is based on the maximum posterior probability value. Bayesian classification is based on Baye's theorem. From Section 2, it is evident that Bayesian classifiers have also exhibited high accuracy and speed when applied to large databases. Naive Bayesian classifiers assume that the effect of a feature value on a given class is independent of the other feature values. This assumtion is called class conditional independence [23]. This work is the extension of Sreeja and Mahalakshmi [24] where emotion is recognized by maximum posterior probability using bag of word feature. In this paper, different set of features and its combinations are used to analyze the emotion recognition process. The Naive Bayes classifier is used to identify emotion from text and conducted this research with a different set of features. The process of identifying emotion from a poem can be defined as follows:

Definition 3:

Let *TP* be a set of training poems contains set of words and their associated emotion labels. Each word is represented by a feature vector, $X_i = (x_1, x_2, x_3 \ldots x_t)$, depicting *t* values made on the token from *t* features, respectively, $x_1,\ x_2,\ \ldots x_t$. Let *p* be a test Poem have X_m words. The emotion classes $E_1, E_2 \ldots E_9$ are love, sadness, courage, anger, hate, joy, fear, peace, surprise. Given testing Poem, *p*, and the classifier will predict that *p* belongs to emotion having highest posterior probability, conditioned on

[4] https://allpoetry.com/poem/11404303-I-Have-Gone-a-Begging-From-Door-to-Door

p. That is, the Naive Bayesian classifier predicts that poem p belongs to the class E_i, if and only if by Equation (2):

$$PR(E_i|p) > PR(E_j|p) \quad for\ 1 \leq j \leq 9, j \neq i\ , \qquad (2)$$

where $p = (X_1, X_{2,\ldots,} X_m)$. Thus maximizes $(E_i|p)$. The class E_i for which $PR(E_i|p)$ is maximized and called the maximum posterior hypothesis. By Bayes' theorem, by Equation (3):

$$PR(E_i \mid p) = \frac{PR(p \mid E_i) \times PR(E_i)}{PR(p)}. \qquad (3)$$

As $PR(p)$ is constant for all classes, it can be maximized using Equation (4):

$$PR(E_i \mid p) = \underset{i \in (1..9)}{\arg\max}\, PR(p \mid E_i) \times PR(E_i)\,, \qquad (4)$$

$PR(E_i)$ is calculated by Equation (5):

$$PR(E_i) = N(E_i) \Big/ \sum_{i=1}^{9} N(E_i)\,, \qquad (5)$$

with many features, it would be computationally expensive to compute $PR(p|E_i)$. To reduce computation, naive assumption of class conditional independence is made by Equation (6). It means that there are no dependency relationships among the features:

$$PR(p \mid E_i) = \prod_{K=1}^{m} (X_K \mid E_i) \qquad (6)$$
$$= PR(X_1 \mid E_i) \times PR(X_2 \mid E_i) \times \ldots\ldots \times PR(X_m \mid E_i).$$

$$PR(X_K \mid E_i) = \prod_{j=1}^{n} (x_{Kj} \mid E_i) \qquad (7)$$
$$= PR(x_{K1} \mid E_i) \times PR(x_{K2} \mid E_i) \times \ldots\ldots \times PR(x_{Kj} \mid E_i).$$

The probabilities $PR(x_{K1}| E_i)$, $PR(x_{K2}| E_i)$, …. $PR(x_{Kj}| E_i)$ are found from the training set, where x_{Kj} refers the K th word's j^{th} feature value.

The main disadvantage of the Naive Bayes classifier is its conditional independence Domingos and Pazzani [25], the assumption that the effect of an attribute value on a given class is independent of the values of other attributes. The other issues are data scarcity [26] and imbalanced classes. To identify the most contributing feature in poem emotion recognition Logistic Regression model is applied.

3.4 Poem Emotion Recognition Using Logistic Regression

In general, regression is a method to predict the value of a dependent variable from one or more independent variables. There are different forms of regression. In this section, the research carried out in logistic regression is discussed. Excel-Solver[5], is

[5] https://www.solver.com/

used to calculate the Logistic Regression Function for each class. The steps involved in calculating Logistic regression function are as follows.

1. Creation of an excel sheet containing numeric feature values.

In Figure[6] 1, column B gives the title of the poem. Columns C to L show the number of nouns, verbs, adjectives, and adverbs; the number of words in the title; the number of repeated words; the number of negation words; and the number of metaphors and similes in the poem. Since the logistic regression is carried out separately for each class, a trick is used, setting the output equal to 1 for the training instances that belong to the class and 0 for those that do not.

2. Sort the dependent variable to make the data evident.

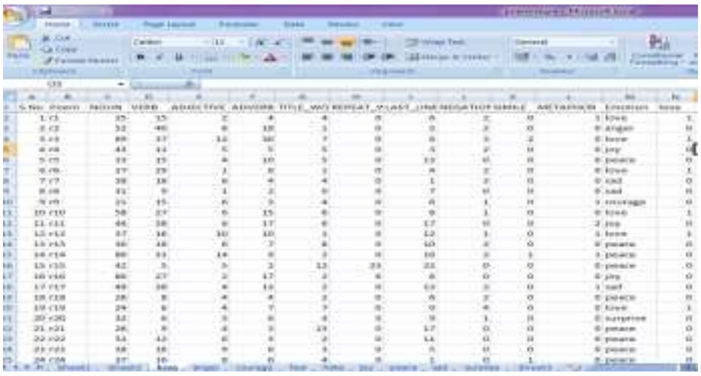
3. From the given input, $x_1, x_2, x_3, \dots\ x_n$ the ‘Logit’ equal to the expression shown on the right-hand side of Equation (8):

$$Logit\ L = b_0 + b_1x_1 + b_2x_2 + \cdots + b_nx_n. \tag{8}$$

The explanatory variables are certain nouns, verbs, adjectives, adverbs, title words, repeating words, last-line words, negation words, similes, and metaphors. The *Logit L* can be written as:

$$\begin{aligned} Logit\ L = b_0 + b_1 noun + b_2 verb + b_3 adjective + b_4 adverb + b_5 title \\ - word + b_6 repeating - word + b_7 last \\ - lineword + b_8 negation + b_9 simile + b_{10} metaphor, \end{aligned}$$

where *Logit L* is a link function states the relation between predictor and the mean of the distribution function, b_0 is the intercept from the regression Equation, b_1 *to* b_{10} are regression coefficients or decision variables. The decision variables $b_0 \dots b_{10}$ are set to 0.1. The solver will adjust the decision variables during the optimization process.



S.No	Poem	NOUN	VERB	ADJECTIVE	ADVERB	TITLE_WO	REPEAT_W	LAST_LINE	NEGATIO	SIMILE	METAPHOR	Emotion	love
1	c1	35	15	2	4	4	0	0	2	0	1	love	1
2	c2	32	40	8	18	1	0	0	2	0	0	anger	0
3	c3	69	37	12	10	7	0	8	3	2	0	love	1
4	c4	44	14	5	5	5	0	5	2	0	0	joy	0
5	c5	19	15	4	10	5	0	13	0	0	0	peace	0
6	c6	27	29	1	8	1	0	4	2	0	0	love	1
7	c7	38	18	6	4	4	0	1	2	0	0	sad	0
8	c8	31	9	1	2	0	0	7	0	0	0	sad	0
9	c9	25	15	6	3	4	0	6	1	0	1	courage	0
10	c10	58	27	6	15	6	0	9	1	0	0	love	1
11	c11	44	38	8	17	6	0	17	0	0	2	joy	0
12	c12	37	16	10	10	3	0	12	1	0	1	love	1
13	c13	66	48	6	7	8	0	10	2	0	0	peace	0
14	c14	80	31	14	9	3	0	10	3	1	1	peace	0
15	c15	42	3	3	2	12	23	22	0	0	0	peace	0
16	c16	88	27	3	17	2	6	6	0	0	0	joy	0
17	c17	48	30	8	10	2	0	13	3	0	1	sad	0
18	c18	28	8	4	4	2	0	6	2	0	0	peace	0
19	c19	24	6	4	7	7	0	0	4	0	0	love	1
20	c20	32	6	3	6	4	0	9	1	0	0	surprise	0
21	c21	26	9	4	3	23	0	17	0	0	0	peace	0
22	c22	34	12	6	6	2	0	14	0	0	0	peace	0
23	c23	38	18	9	8	3	0	5	0	0	0	peace	0
24	c24	37	16	8	6	4	0	1	0	1	0	peace	0

Fig. 1. Data for Logistic Regression Model.

[6] http://dx.doi.org/10.17632/n9vbc8g9cx.1

4. Calculate the $\boldsymbol{e^L}$ for each record and *PR(X)* by Equation (9):

where *'e'*=2.718281.

$$PR(X) = e^L/(1 + e^L), \tag{9}$$

where $Logit\ L = b_0 + b_1x_1 + b_2x_2 + \cdots + b_nx_n$.

5. Calculate the Log-likelihood Function (LL) using Equation (10):

$$PR(Y_i = y_i|x_{1i}, x_{2i}, \ldots. x_{ni}). \tag{10}$$

Equation (10) is the conditional probability that the predicted dependent variable y_i equals the observed value Y_i for the given independent variables $x_{1i}, x_{2i}, \ldots. x_{ni}$. It can be calculated by the following Equation (11):

$$PR(Y = y|X) = PR(X)^y[1 - PR(X)]^{(1-y)}. \tag{11}$$

Taking natural logarithm on both sides of Equation (11) leads to Equation (12):

$$\ln[PR(Y = y|X)] = y.\ln[PR(X)].(1 - y)\ln[1 - PR(X)]. \tag{12}$$

Log-likelihood function LL is the sum of terms for all data records by the following Equation (13):

$$LL = \sum Y_iPR(X_i) + (1 - Y_i)(1 - PR(X_i)). \tag{13}$$

6. Calculate the *Maximum Log-Likelihood (MLL)* function using Solver.

The objective of the Logistic regression is to find the decision variables that maximize the *LL* function to produce the *MLL*. The Solver adjusts the decision variables GRG Nonlinear solving method. Solver was run several trials to obtain the optimum solution. The optimal solution is calculated by equation (14) to (22) for each emotion category.

$$Love = 0.3736 + 0.0042x_1 - 0.0056x_2 - 0.026x_3 + 0.0063x_4 + 0.0023x_5 + 0.0036x_6 - 0.0091x_7 - 0.0121x_8 + 0.0157x_9 + 0.0871x_{10} \tag{14}$$

$$Anger = 0.0137 + 0.0003x_1 + 0.0008x_2 + 0.0005x_3 + 0.0004x_4 - 0.0011x_5 - 0.0004x_6 - 0.0013x_7 - 0.0001x_8 + 0.0055x_9 + 0.0022x_{10} \tag{15}$$

$$Courage = 0.0402 + 0.0007x_1 - 0.0014x_2 - 0.0008x_3 + 0x_4 + 0.0026x_5 - 0.002x_6 + 0.0011x_7 + 0.0034x_8 - 0.005x_9 - 0.0069x_{10} \tag{16}$$

$$Fear = 0.0114 + 0.0003x_1 + 0.0012x_2 - 0.0006x_3 - 0.0015x_4 - 0.0013x_5 + 0.0005x_6 - 0.0003x_7 - 0.0006x_8 + 0.0045x_9 + 0.0049x_{10} \tag{17}$$

$$Hate = 0.0062 + 0.0002x_1 - 0.0001x_2 - 0.001x_3 + 0.0001x_4 + 0.0001x_5 - 0.0001x_6 - 0.0003x_7 - 0.0008x_8 - 0.0019x_9 - 0.001x_{10} \tag{18}$$

$$Joy = 0.1358 + 0.0001x_1 + 0.0012x_2 + 0.0026x_3 - 0.0084x_4 + 0.0026x_5 + 0.0013x_6 + 0.0088x_7 - 0.0045x_8 - 0.0003x_9 - 0.0047x_{10} \quad (19)$$

$$Peace = 0.1969 - 0.0024x_1 + 0.0044x_2 + 0.0149x_3 - 0.0059x_4 - 0.0008x_5 - 0.0016x_6 - 0.0009x_7 + 0.0044x_8 - 0.0374x_9 - 0.0555x_{10} \quad (20)$$

$$Sadness = 0.2168 - 0.0031x_1 + 0.0004x_2 + 0.0104x_3 + 0.0087x_4 - 0.0025x_5 - 0.0006x_6 - 0.001x_7 + 0.0066x_8 + 0.037x_9 - 0.029x_{10} \quad (21)$$

$$Surprise = 0.0324 + 0.0002x_1 - 0.0008x_2 - 0.0001x_3 + 0.0004x_4 - 0.002x_5 - 0.0009x_6 + 0.0011x_7 + 0.0022x_8 - 0.007x_9 + 0.0074x_{10} \quad (22)$$

From these equations, it is observed that metaphors contribute the most to identifying emotions like love, fear, and surprise. Further, it is noted that similes portray sadness, verbs display anger, and negation words reveal courage. Nouns illustrate hate, last-line words depict joy and adjectives evoke peace. The result of this prediction experiment is given below. For a given test poem, if the emotion to be predicted is love, then, of the results obtained by Equations (14) to (22), the output of Equation (14) should be greater than all the other values.

4 Results and Discussions

The main objective is to evaluate the importance of the features described in Section 3.2 for recognizing the emotion of a poem. To evaluate these features, the experiments are carried out using the PERC, by considering features in single and combined mode. These methods are experimented with in 10-fold cross-validation. In 10-fold cross-validation [27], the data set is split into ten sets of size n=10; train on nine datasets and test on 1. This process is repeated ten times to achieve better accuracy. Another key reason for using ten-fold cross-validation instead of conventional validation is if there is insufficient data to be partitioned into two distinct training and testing sets without missing out on unique information. In such cases, cross-validation is a suitable technique.

Table 3 shows that 443 poems are correctly classified using linguistic features alone. Nouns, verbs, adjectives, and adverbs are the linguistic features included in the model. Table 4 shows that 484 poems are correctly classified using only poetic features, similes, and metaphors.

Table 5 shows that 503 poems are classified correctly when a combination of poetic and linguistic features is used. Table 6 shows that 535 poems are identified correctly when all the features are used for classification. A consistent improvement in results is observed, and Tables 3, 4, 5, and 6 show that the classification carried out with all the features (linguistic, semantic, orthographic, poetic and statistical) gives better results than the other methods. Table 7 shows that 560 poems are identified correctly by this method.

Table 3. Confusion Matrix of 10-fold Cross Naive Bayes Classifier with Linguistic Features.

Emotion	No. of Poems in PERC	Anger	Courage	Fear	Hate	Joy	Love	Peace	Sad	Surprise
Anger	52	30	8	0	7	0	0	0	7	0
Courage	64	7	32	0	5	8	5	0	3	4
Fear	54	3	0	20	9	2	9	1	10	0
Hate	54	13	5	3	22	0	1	0	9	1
Joy	100	1	9	0	1	68	14	2	0	5
Love	154	2	4	0	2	8	115	3	18	2
Peace	82	1	4	3	0	8	7	50	4	5
Sad	126	4	0	5	7	0	15	3	92	0
Surprise	50	3	4	4	1	13	5	6	0	14

Table 4. Confusion Matrix of 10-fold Cross Naive Bayes Classifier with Poetic Features.

Emotion	No.of Poems in PERC	Anger	courage	Fear	Hate	Joy	Love	Peace	Sad	Surprise
Anger	52	32	7	0	7	0	0	0	6	0
Courage	64	5	34	1	5	6	5	1	3	4
Fear	54	3	0	24	9	2	9	0	7	0
Hate	54	10	4	5	20	0	4	0	8	3
Joy	100	1	8	1	1	65	16	2	1	5
Love	154	2	3	0	1	5	120	1	21	1
Peace	82	3	4	2	0	5	7	52	4	5
Sad	126	2	0	1	1	0	12	0	110	0
Surprise	50	1	2	2	1	8	4	5	0	27

Table 5. Confusion Matrix of 10-fold Cross Naive Bayes Classifier with Linguistic and Poetic Features.

Emotion	No. of Poems in PERC	Anger	Courage	Fear	Hate	Joy	Love	Peace	Sad	Surprise
Anger	52	34	4	0	6	0	1	0	7	0
Courage	64	4	38	0	4	5	4	0	4	5
Fear	54	2	0	21	10	2	11	0	8	0
Hate	54	7	4	6	25	0	2	0	7	3
Joy	100	1	7	0	0	67	18	2	1	4
Love	154	2	5	0	2	3	120	3	18	1
Peace	82	3	4	2	0	4	5	57	4	3
Sad	126	2	0	1	2	0	9	0	112	0
Surprise	50	1	3	2	2	10	5	4	0	23

Table 6. Confusion Matrix of 10-fold Cross Naive Bayes Classifier with Linguistic, Orthographic, Poetic, Statistical and Semantic Features.

Emotion	No. of Poems in PERC	Anger	Courage	Fear	Hate	Joy	Love	Peace	Sad	Surprise
Anger	52	40	6	0	3	0	0	0	3	0
Courage	64	6	42	0	3	7	3	0	1	2
Fear	54	1	0	30	7	0	7	0	9	0
Hate	54	11	3	2	32	0	0	0	6	0
Joy	100	0	8	0	0	78	11	0	0	3
Love	154	0	1	0	0	6	130	1	16	0
Peace	82	1	3	1	0	7	5	60	2	3
Sad	126	2	0	2	5	0	12	3	102	0
Surprise	50	2	3	5	0	9	7	3	0	21

Table 7. Confusion Matrix of 10-fold Cross Logistic Regression.

Emotion	No.of Poems	Anger	courage	Fear	Hate	Joy	Love	Peace	Sad	Surprise
Anger	52	42	1	4	3	0	1	0	1	0
Courage	64	4	36	0	4	6	5	2	2	5
Fear	54	0	0	42	2	0	4	0	4	2
Hate	54	7	4	3	34	0	2	1	6	1
Joy	100	3	5	0	0	78	6	5	1	2
Love	154	0	0	0	0	5	137	3	7	2
Peace	82	1	1	3	0	7	7	60	1	2
Sad	126	2	1	2	5	0	8	2	105	1
Surprise	50	1	2	4	0	10	4	3	0	26

Table 8. Comparison of Precision Measures.

Emotion	All Features	Linguistic Features	Poetic Features	Linguistic and Poetic Features	Logistic Regression
Anger	0.635	0.469	0.542	0.607	0.69
Courage	0.636	0.485	0.548	0.585	0.75
Fear	0.75	0.571	0.667	0.711	0.652
Hate	0.64	0.407	0.444	0.49	0.696
Joy	0.729	0.636	0.714	0.736	0.736
Love	0.743	0.673	0.678	0.686	0.778
Peace	0.896	0.769	0.852	0.864	0.789
Sad	0.734	0.643	0.688	0.696	0.823
Surprise	0.724	0.452	0.6	0.59	0.583

Table 8 gives detailed information on the precision measures obtained in the four experiments. The experiment with all the features gives a more promising precision measure when compared to the other three methods. The emotion category, peace, has a higher precision measure compared to the other classes. Table 9 details the recall measures obtained in the four experiments. The experiment with all the features gives a more promising recall measure, except in the case of the emotion category, sadness. The experiment with linguistic and poetic features gives better results in the classification of sad poems. Table 10 shows that the experiment with all the features offers better results compared to that of linguistics and poetic features. However, in the classification of sad poems, the experiment with all the features as well as the experiment with the linguistic and poetic features combined give, more or less, the same F-measure.

Table 9. Comparison of Recall Measures.

Emotion	All Features	Linguistic Features	Poetic Features	Linguistic and Poetic Features	Logistic Regression
Anger	0.769	0.577	0.615	0.654	0.8
Courage	0.656	0.5	0.531	0.594	0.6
Fear	0.556	0.37	0.444	0.5	0.714
Hate	0.593	0.407	0.37	0.463	0.615
Joy	0.78	0.68	0.65	0.67	0.78
Love	0.844	0.747	0.779	0.779	0.884
Peace	0.732	0.61	0.634	0.695	0.732
Sad	0.81	0.73	0.873	0.889	0.829
Surprise	0.42	0.28	0.54	0.46	0.467

Table 10. Comparison of F Measures.

Emotion	All Features	Linguistic Features	Poetic Features	Linguistic and Poetic Features	Logistic Regression
Anger	0.696	0.517	0.577	0.63	0.741
Courage	0.646	0.492	0.54	0.589	0.667
Fear	0.638	0.449	0.533	0.587	0.682
Hate	0.615	0.407	0.404	0.476	0.653
Joy	0.754	0.657	0.681	0.702	0.757
Love	0.79	0.708	0.725	0.729	0.828
Peace	0.805	0.68	0.727	0.77	0.759
Sad	0.77	0.684	0.769	0.78	0.826
Surprise	0.532	0.346	0.568	0.517	0.519

5 Conclusion

In this research, the importance of features in emotion identification is studied. The proposed work, also groups the features and evaluated the significance of each group of features in terms of the results. It is found that certain features - linguistic, poetic, statistical, semantic and orthographic - influence the results significantly. Accuracy was somewhat diminished, given the shortcomings of the Naive Bayes classifier in terms of conditional independence, data scarcity, and imbalanced classes. Laplace smoothing solved information scarcity. Experiments with the Logistic Regression

Method helped determine the contribution of each feature in the process of prediction. As future work, we intend to increase the corpus size and number of poetic features such as meter, sarcasm, and other poetic devices. To improve the accuracy some ensemble of base classifier can be applied.

References

1. Rangacharya, A.: Bharatha Natyshastra. Munshiram Manoharlal Publishers (1996)
2. Ghosh, M.: The natyashastra (English translation) volume i (chapters i-xxvii). Calcutta: The Royal Asiatic Society of Bengal (1950)
3. Alm, E.C.O.: Affect in text and speech. University of Illinois at Urbana Champaign (2008)
4. Wu, C.H., Chuang, Z.J., Lin, Y.C.: Emotion recognition from text using semantic labels and separable mixture models. ACM transactions on Asian language information processing (TALIP), 5(2), 165–183 (2006)
5. Strapparava, C., Mihalcea, R.: Learning to identify emotions in text. In: Proceedings of the 2008 ACM symposium on Applied computing, ACM, 1556–1560 (2008)
6. Ekman, P.: An argument for basic emotions. Cognition & emotion 6(3-4), 169–200 (1992)
7. Minato, J., Bracewell, D.B., Ren, F., Kuroiwa, S.: Japanese emotion corpus analysis and its use for automatic emotion word identification. Engineering Letters, 16(1) (2008)
8. Das, D., Bandyopadhyay, S.: Word to sentence level emotion tagging for bengali blogs. In: Proceedings of the ACL-IJCNLP 2009 Conference Short Papers, Association for Computational Linguistics, 149–152 (2009)
9. Perfors, A., Tenenbaum, J.B., Griffiths, T.L., Xu, F.: A tutorial introduction to bayesian models of cognitive development. Cognition, 120(3), 302–321 (2011)
10. Bielza, C., Li, G., Larranaga, P.: Multi-dimensional classification with Bayesian networks. International Journal of Approximate Reasoning, 52(6), 705–727 (2011)
11. Yoon, H.J., Chung, S.Y.: Eeg-based emotion estimation using bayesian weightedlog-posterior function and perceptron convergence algorithm. Computers in biology and medicine 43(12), 2230–2237 (2013)
12. Lei, J., Rao, Y., Li, Q., Quan, X., Wenyin, L.: Towards building a social emotion detection system for online news. Future Generation Computer Systems, 37, 438–448 (2014)
13. Lee, S.H., Yang, K.M., Cho, S.B.: Integrated modular bayesian networks with selective inference for context-aware decision making. Neurocomputing, 163, 38–46 (2015)
14. Steyerberg, E.W., Eijkemans, M.J., Harrell Jr, F.E., Habbema, J.D.F.: Prognostic modeling with logistic regression analysis: in search of a sensible strategy in small data sets. Medical Decision Making, 21(1), 45–56 (2001)
15. Dmello, S.K., Craig, S.D., Witherspoon, A., Mcdaniel, B., Graesser, A.: Automatic detection of learners affects from conversational cues. User modeling and user adapted interaction, 18(1), 45–80 (2008)
16. Fleiss, J.L.: Measuring nominal scale agreement among many raters. Psychological bulletin, 76(5), 378 (1971)
17. Lubis, N., Lestari, D., Purwarianti, A., Sakti, S., Nakamura, S.: Construction and analysis of Indonesian emotional speech corpus. In: Coordination and Standardization of Speech Databases and Assessment Techniques (COCOSDA), 2014 17th Oriental Chapter of the International Committee for the IEEE, 1–5 (2014)
18. Toutanova, K., Klein, D., Manning, C.D., Singer, Y.: Feature-rich part-of-speech tagging with a cyclic dependency network. In: Proceedings of the 2003 Conference of the North

American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1, Association for Computational Linguistics, 173–180 (2003)

19. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Communications of the ACM, 18(11), 613–620 (1975)
20. Liu, H., Singh, P.: Conceptneta practical commonsense reasoning tool-kit. BT technology journal, 22(4), 211–226 (2004)
21. Liu, H., Singh, P.: Commonsense reasoning in and over natural language. In: Knowledge-based intelligent information and engineering systems, Springer, 293–306 (2004)
22. Sreeja, P.S., Mahalakshmi, G.S.: Concept identification from poems. In: Recent Trends and Challenges in Computational Models (ICRTCCM), 2017 Second International Conference on IEEE, 211–216 (2017)
23. Tan, P.N., et al.: Introduction to data mining. Pearson Education India (2006)
24. Sreeja, P.S., Mahalakshmi, G.S.: Emotion recognition from poems by maximum posterior probability. International Journal of Computer Science and Information Security 14, (CIC 2016), 36–43 (2016)
25. Domingos, P., Pazzani, M.: On the optimality of the simple bayesian classifier under zero one loss. Machine learning, 29(2), 103–130 (1997)
26. Rennie, J.D., Shih, L., Teevan, J., Karger, D.R.: Tackling the poor assumptions of naive bayes text classifiers. In: Proceedings of the 20th International Conference on Machine Learning (ICML-03), 616–623 (2003)
27. Kohavi, R., et al.: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: Ijcai, Volume 14, Stanford, CA, 113–1145 (1995)

ISSN 1870-4069

Looking Relationships between Pre-admission Factors and Students' Performance Using Association Rules

Silvia Beatriz González-Brambila, Josué Figueroa-González

Universidad Autónoma Metropolitana,
Azcapotzalco, Mexico

{sgb,jfgo}@correo.azc.uam.mx

Abstract. High school institutions are very demanded, thousands of students desire entering to them and universities are looking for the best candidates considering several criteria like admission tests, former level average or extra curricular activities. Educational Data Mining is a branch of Data Mining which processing a lot of educative data, allows finding interesting patterns which allow taking decisions for solving or improving educative aspects. This work presents the use of Data Mining Association Rules technique for finding rules related with students pre-admission characteristics and their first scholar year performance. Goal of the paper is analyzing if using as single criteria the highest mark in an admission test allows the Mexican Autonomous Metropolitan University choosing students which will have the best performance. Results show that even if admission test mark is relevant, there exist other factors, specially the former level average, which should be considered in the admission process.

Keywords. Association rules, educational data mining, enrollment criteria, student performance, university admission.

1 Introduction

Universities use different enrollment criteria for choosing students that will enter to a superior education. Entrance examination, personal interviews, scholar performance on the previous levels, extracurricular activities, etc, are some of the criteria considered for accepting a student. The admission process looks up for choosing the best students, so they can finish their studies with an acceptable performance; however it's not clear if these criteria allow universities to make it.

With the increase in the use of technology, it's possible to store a lot of data from students, which being processed can lead to interesting information that can be helpful for making decisions. Data Mining (DM) is the process of analyzing large amount of data for discovering patterns that lead to knowledge [6], applying this process to academic data is known as Educational Data Mining (EDM) [2].

DM offers a lot of algorithms according a desired goal; in particular association techniques allow discovering relationships among several data. Association rules show the relationship between certain event or events called antecedents, and another event, or events, called consequence.

In this work is applied the Association Rules technique for discovering relationships between some entrance criteria and the performance of engineering students over their first year at the Autonomous Metropolitan University (UAM). This one, uses as enrollment criteria a single admission exam where students with best marks are considered for entering university. This mark must be higher or at least, very near, to 600 points (from a maximum of 1000), other prerequisite is that medium high level average, in Mexico is the former level to superior education, must be at least 7.0 (from a maximum of 10). Consider that not all the students which fulfill these criteria are accepted, only a small number of candidates, for this reason it's important to know if a single admission test is the best way for choosing from all the students that look up for entering UAM.

At UAM, first year is called the General Branch Level (GBL) and is composed by 3 quarters (scholar period at UAM) where all engineering students study the same topics and it's the period with the highest level of desertion and the one which take students most time for finishing it. For these reasons, it's performed a study than can find relationships between some criteria besides the admission test and the performance of students.

Goal of this work is analyzing the rules obtained from the relationship between the single entrance criteria, admission test, and other which now are not considered like medium high school average, student age and school of origin with the academic performance of students during their first scholar year. Performance is measured using Grade Point Average (GPA) and the amount of approved credits during this period. GPA is a measure obtained from the relationship of approved topics multiplied and the obtained mark against attempted credits and it's obtained using Eq. (1):

$$GPA = \frac{\sum(credits \ * \ mark)}{attempted \ credits}, \tag{1}$$

where:

- credits is the amount of credits of a subject,
- mark is the obtained mark in a topic (only for approved ones),
- attempted credits is the total of credits coursed, approved or not, in a certain period.

This will allow determining if there is a relationship between student performance and entrance exam, or if it's necessary to consider other factors that actually are being ignored in the process of choosing students at UAM.

Paper is structured as follows: Section 2 presents related works about the topic. Section 3 presents the processes of obtaining data and generating the association rules. Section 4 shows the analyzis of the obtained rules for both performances, considering GPA and amount of approved credits. Section 5 presents Conclusions and Future works.

2 Related Work

The main topic studied with EDM is student performance over a certain period or for all studies, related with student performance over first year is [4], where linear regression and Decisions Trees are used for predicting one of three categories created according students first year performance over a set of the French Community of Belgium. Results show that academic factors like average in last year of secondary school, time of studying mathematics and class attendance have a strong relationship with the classification of a student in one of the three categories based in its performance.

Besides this topic, several works related with the analyzis of enrollment criteria over students' performance have been performed. This is an interesting topic for universities because they are always looking for the best way of choosing their new students.

In [8], the relationships between several factors including age, GPA of previous studies, admission test marks and performance over first semesters in a former level with the first semester in a nursing school program are obtained. Used technique is regression analysis, results shown that admission test mark is the criteria which has most relevance with student performance.

In [9], was performed a study about the influence of academic and personal factors over first semester and first year academic performance in an Ethiopian university. Used techniques were correlation analysis and linear regression. Results shown that former grade and admission test have the greatest influence over student performance, also socio economic variables has certain relationship with student achievement.

Finally, in [3], the importance of general tests, called State Matura exams, and the average in former level have over the performance of student in two faculties in Croatia is studied. Matura exams include Croatian language, mathematics and a foreign language. Used techniques are also multiple linear regression and correlation analysis. Obtained results are not considered statistical relevant and it's concluded that Matura exams are not useful for predicting students performance.

As can be seen, all of the analyzed works used the correlation analysis and multiple regressions for predicting the performance of students. In this work, association rules are used, not for predicting, but for identifying relationships among several factors with student performance.

3 Applying Data Mining for Discovering Rules

Considering the steps of several DM methodologies [1], once the goal is defined, the process for obtaining knowledge must include at least the following steps: obtaining and preparing data, applying DM algorithms, in this case Association Rules algorithm, and finally, analyzing results for obtaining knowledge.

3.1 Obtaining and Preparing Data

UAM stores student data into two sources, the General File of Students (AGA for its acronym in Spanish) which contains a lot of personal and academic data from all the students in the university. Second source is the academic record, called kardex in UAM, these data correspond specifically to the academic history of every student, each register is composed by the registration number of a student, id of a topic, obtained mark and the period when the student studied the topic. For this work, were considered Computer Engineering students from years 2010 to 2016 for a total of 538 students. Students from previous years were not considered because their marks, in 2010, a major modification in the students' plan was performed, so marks in several subjects were change to EQ (Equivalent) and the original mark wasn't obtained.

From the AGA, were obtained the following data related with pre-admission characteristics:

- School of Origin (SCH). Indicates the medium high level school from where a student proceeds.
- Medium high school average (AVG). Represents the average obtained by a student in the medium high level.
- Entrance age (AGE). Is the age of the student at the moment of being accepted into the university.
- Admission exam mark (PEX). Total of points earned by a student in the admission exam that UAM applies for candidates of their engineering programs.

From kardex, were obtained the studied topics and obtained marks from the first year, in UAM, the scholar calendar is divided in quarters, three quarters per year, so, were considered the topics of the first three quarters according the entrance quarter (university has two periods for entrance). Marks at UAM are assigned with a letter, these are presented in Table 1; letters are presented according their acronym in Spanish. While Not Approved mark stands for 5, at the moment of obtaining GPA, the used value was 0.

Table 1. Marks and their numeric equivalent at UAM.

Mark	Numeric Equivalent
MB (Very Good)	10
B (Good)	8
S (Sufficient)	6
NA (Not Approved)	5

Data preparation, or data cleaning, involves obtaining GPA and the amount of credits approved for each student during its first year. For obtaining the amount of credits, were added the credits of each approved topic during the first year (three quarters) of every student. Credits of each topic were taken from the study plan of Computing Engineering [7].

For obtaining better results using the Association Results technique, data were categorized, so, were created categories for admission test, medium high school average, student age, school of origin, GPA and amount of approved credits. Next, are presented the used categories, in each one, between parentheses, values that appears in the rules.

For creating admission test categories were considered the range of values presented in Table 2, consider that the highest value was 909 (from a maximum of 1000) and the lowest, 548 points.

Table 2. Categories of admission test obtained mark.

Range of Values	Equivalent
More or equal than 880	High (EX_HIGH)
From 770 to 880	Medium (EX_MED)
Less than 770	Low (EX_LOW)

Categories used for medium high school average are presented in Table 3. These categories were created considering the most common marks scale used at UAM. Here the values run from 7.0 to 10, remembering that 7.0 is the minimum value for accessing the university.

Table 3. Categories of medium high school average.

Range of Values	Equivalent
More or equal than 8.8	High (AVG_HIGH)
From 7.7 to 8.8	Medium (AVG_MED)
Less than 7.7	Low (AVG_LOW)

Categories for entrance age are presented in Table 4. Ages from accepted students were from 16 to 41 years old. The most common entrance age in Mexico to superior level runs from 18 to 20 years old.

Table 4. Categories of entrance age.

Range of Values	Equivalent
Less than 18	Group 1 (GPR_1)
From 18 to 20	Group 2 (GPR_2)
From 20 to 25	Group 3 (GPR_3)
Greater than 25	Group 4 (GPR_4)

Categories of GPA are presented in Table 5. Obtained GPA of processed students runs from 0.0 (meaning that a student didn't approve any topic) to 9.9.

Table 5. Categories from Grade Point Average.

Range of Values	Equivalent
More or equal than 8.8	High (GPA_HIGH)
From 7.6 to 8.7	Medium (GPA_MED)
From 6.0 to 7.5	Low (GPA_LOW)
From 0.0 to 5.9	Very Low (GPA_VLOW)

Finally, categories for approved credits are presented in Table 6. The expected amount of credits considering the Computing Engineering study plan for the first year is 105; however, at UAM, students can study their topics in a different order than the presented in the study plan, they even can approve more credits; for this reason, were considered the estimated amount of credits expected for quarter for the classification: 1st quarter 32 credits, 2nd quarter 33 and 3rd quarter, 40. Credits for analyzed students run from 0 to 120 credits.

Table 6. Categories from approved credits.

Range of Values	Equivalent
From 80 to more than 105	High (CRE_HIGH)
From 32 to less than 80	Medium (CRE_MED)
Less than 32	Low (CRE_LOW)

This classification means that a student with a low performance, approved less than 32 credits in its first year (less than the credits of 1st quarter), a medium performance represents approving from 32 to 80 credits (the sum of credits of first, second quarter and some credits of third quarter) during the first year. Finally, the high performance is from 80 to more than 105, that represents approving all, or almost, the topics from first, second and third quarter during the first year.

Also, was considered the school of origin of every student, schools were grouped according the main institution system they belong. However, for privacy reasons, in this work are presented with generic names, School A (SCH_A) to School I (SCH_I).

3.2 Obtaining Association Rules

For generating Association Rules, was used the *Apriori* algorithm; with this algorithm, many rules are generated, so it's necessary to choose the most representative, there are several methods for choosing the best rules [5], the most used value for this is lift, which represents the occurrence frequency of a set of elements X and Y respecting an expected value and is defined in Eq. (2):

$$lift(X \Rightarrow Y) = \frac{support(X \Rightarrow Y)}{support(X) \, * \, support(Y)}, \tag{2}$$

where support is defined as the percentage of transactions that contains X in a set of transactions D.

For evaluating the importance of a rule, are considered the following criteria: if value of lift is equal to 1, means that the relationship could be due to a random consequence. If lift is greater than 1, represents that exists a strong relationship among antecedents and consequences (X and Y are called complements). Finally, if lift is less than 1, there is not a relationship (X and Y are called substitutes).

Were obtained rules considering as antecedents different combinations of values for admission test, medium high school average, student age and school of origin, consequence events were GPA and amount of approved credits separately. Rules for GPA are presented in Tables 7 to 9 and the ones for total of credits are shown in Tables 10 to 12.

Table 7. Rules obtained for high performance measured with GPA.

Rule	Support	Confidence	Lift
{EX_HIGH,GPR_2,SCH_I} ⇒ {GPA_HIGH}	0.001879	1.0	13.641
{AVG_MED,EX_HIGH,GPR_2} ⇒ {GPA_HIGH}	0.001879	1.0	13.641
{AVG_HIGH,SCH_D,EX_LOW} ⇒ {GPA_HIGH}	0.001879	1.0	13.641
{AVG_HIGH,EX_LOW,GPR_3,SCH_E} ⇒ {GPA_HIGH}	0.001879	1.0	13.641
{AVG_HIGH,SCH_D,GPR_2} ⇒ {GPA_HIGH}	0.005639	0.75	10.230

4 Analyzing Association Rules

Next are presented the analysis of the generated rules for each case.

4.1 Interpreting Rules Related with Performance Measured Using GPA

Analyzing the antecedents that tends to a high performance considering GPA, rules from Table 7, it's clear that a high mark in the admission exam and a high average in former level (EX_HIGH) and AVG_HIGH) are associated with a higher performance, this can be a rule called trivial, however, it's interesting that in two rules exist a low mark in the admission exam (EX_LOW) that appears with a high average. Talking about entrance age, two groups can be found in the rules, the group of the most common entrance age (GPR_2) and another group which age is little older (GPR_3).

Table 8. Rules obtained for medium performance measured with GPA.

Rule	Support	Confidence	Lift
{EX_HIGH,SCH_G} ⇒ {GPA_MED}	0.001879	1.0	3.5231
{SCH_D,EX_MED,GPR_4} ⇒ {GPA_MED}	0.001879	1.0	3.5231
{AVG_MED,SCH_D,EX_MED} ⇒ {GPA_MED}	0.001879	1.0	3.5231
{AVG_MED,EX_MED,SCH_H} ⇒ {GPA_MED}	0.001879	1.0	3.5231
{EX_MED,GPR_3,SCH_H} ⇒ {GPA_MED}	0.001879	1.0	3.5231
{SCH_C,EX_MED,GPR_2} ⇒ {GPA_MED}	0.001879	1.0	3.5231
{AVG_MED,SCH_C,EX_MED} ⇒ {GPA_MED}	0.001879	1.0	3.5231
{AVG_HIGH,SCH_C,GPR_4} ⇒ {GPA_MED}	0.001879	1.0	3.5231
{EX_MED,GPR_3,SCH_I} ⇒ {GPA_MED}	0.001879	1.0	3.5231
{AVG_LOW,SCH_A,EX_MED} ⇒ {GPA_MED}	0.001879	1.0	3.5231
{AVG_LOW,SCH_A,GPR_4} ⇒ {GPA_MED}	0.001879	1.0	3.5231
{AVG_MED,EX_MED,GPR_4} ⇒ {GPA_MED}	0.001879	1.0	3.5231
{AVG_LOW,EX_MED,SCH_F} ⇒ {GPA_MED}	0.001879	1.0	3.5231
{AVG_LOW,SCH_C,EX_LOW,GPR_3} ⇒ {GPA_MED}	0.001879	1.0	3.5231
{SCH_A,EX_MED,GPR_3} ⇒ {GPA_MED}	0.005639	0.75	2.6423

Table 9. Rules obtained for low performance measured with GPA.

Rule	Support	Confidence	Lift
{GPR_4,SCH_H} ⇒ {GPA_LOW}	0.001879	1.0	3.4322
{AVG_LOW,GPR_4,SCH_I} ⇒ {GPA_LOW}	0.003759	1.0	3.4322
{EX_MED,SCH_F,GPR_4} ⇒ {GPA_LOW}	0.001879	1.0	3.4322
{AVG_LOW,EX_MED,GPR_2} ⇒ {GPA_LOW}	0.001879	1.0	3.4322
{AVG_MED,EX_MED,GPR_2,SCH_G} ⇒ {GPA_LOW}	0.001879	1.0	3.4322
{EX_LOW,GPR_4,SCH_I} ⇒ {GPA_LOW}	0.0075187	0.8	2.7458
{AVG_LOW,GPR_3,SCH_I} ⇒ {GPA_LOW}	0.0075187	0.8	2.7458

After reviewing rules in Table 8, the ones related with a medium performance considering GPA, they show something interesting, the average of the former level it's medium (AVG_MED), there is a set of rules containing high former level average (AVG_HIGH), but appears with the oldest group of age (GPR_4) which doesn't appear in the high GPA performance. Also, in this set of rules, doesn't appear neither high (EX_HIGH) not low (EX_LOW) marks in the admission test.

Analyzing the seven rules in Table 9, associated with a low performance, can be seen that five of them contain one of the groups (GPR_3 and GPR_4) which are older than the common entrance age. Also the former level average is classified in low or medium (AVG_MED and AVG_LOW). About admission exam, only were found medium marks (EX_MED).

From this analysis, can be told that the average in the former level, in this case, the medium high level, can lead to choose students which trend to have a better performance. The fact that only one high or low admission exam marks and none high or low former average appears in the rule of medium performance, strengths the theory that a high performance can be associated with a high mark

Table 10. Rules obtained for high performance measured with amount of credits.

Rule	Support	Confidence	Lift
{EX_HIGH,GPR_2} ⇒ {CRE_HIGH}	0.003759	1.0	2.67336
{EX_MED,SCH_I} ⇒ {CRE_HIGH}	0.01357	1.0	2.67336
{AVG_HIGH,SCH_C,GPR_4} ⇒ {CRE_HIGH}	0.001879	1.0	2.67336
{AVG_HIGH,GPR_2,SCH_E} ⇒ {CRE_HIGH}	0.001879	1.0	2.67336
{AVG_HIGH,EX_LOW,SCH_E} ⇒ {CRE_HIGH}	0.003759	1.0	2.67336
{EX_MED,GPR_2} ⇒ {CRE_HIGH}	0.03571	0.8260	2.20843
{AVG_HIGH,SCH_D} ⇒ {CRE_HIGH}	0.007518	0.8	2.13869
{AVG_HIGH,EX_MED} ⇒ {CRE_HIGH}	0.04699	0.7575	2.02527
{AVG_LOW,GPR_2,SCH_I} ⇒ {CRE_HIGH}	0.005639	0.75	2.00502
{AVG_HIGH,SCH_A,GPR_4} ⇒ {CRE_HIGH}	0.005639	0.75	2.00502
{EX_MED,SCH_E} ⇒ {CRE_HIGH}	0.0206	0.7333	1.96046
{AVG_HIGH,SCH_I} ⇒ {CRE_HIGH}	0.0244	0.7222	1.93076

in the admission test and a high average in the medium high level.

As can be seen, the entrance age also plays an important role in the performance of the student, showing that groups with an older age, specially Group 4, tends to have a lower performance.

4.2 Interpreting Rules Related with Performance Considering the Mmount of Approved Credits

Can be considered that as highest is the GPA, highest is the amount of credits, however there it's not relationship, for example: a GPA of 10 (the maximum value) can be obtained approving one topic with the highest mark (MB), and a GPA of 6 can be obtained approving more than one topic with the minimum approving mark (S). Also, it's supposed that the characteristics associated with a high performance considering GPA, should be the same for a high performance considering the amount of credits, but to assure this, the set of rules related with the amount of credits are also analyzed.

As rules associated with high amount of approved credits, Table 10, show, having a high average in the former level (AVG_HIGH) appears in most of the rules with the highest value of lift. Here the relationship of the mark in the admission exam it's not as clear as with the performance considering GPA. Were obtained rules with each one of the possible values for the exam (EX_HIGH, EX_MED and EX_LOW). Bigger amount of credits it's associated also with several entrance age groups, but it's clear that the common entrance age group (GPR_2) is present in most of the rules, even with an entrance exam mark or former level average which are not high, while another group (GPR_4) is present only with high average in medium high level.

Looking the rules of Table 11, the ones related with medium performance, it's clear that former level average has an important role in the amount of accumulated credits, most of the rules contain low or medium average (AVG_LOW and AVG_MED). Also the mark in the admission exam has many low and medium

Table 11. Rules obtained for medium performance measured with amount of credits.

Rule	Support	Confidence	Lift
{EX_HIGH,GPR_3} ⇒ {CRE_MED}	0.003759	1.0	2.1111
{SCH_D,GPR_3} ⇒ {CRE_MED}	0.007518	1.0	2.1111
{SCH_D,EX_MED,GPR_4} ⇒ {CRE_MED}	0.001879	1.0	2.1111
{AVG_LOW,GPR_3,SCH_I} ⇒ {CRE_MED}	0.009398	1.0	2.1111
{AVG_LOW,SCH_A,GPR_4} ⇒ {CRE_MED}	0.001879	1.0	2.1111
{EX_MED,GPR_4,SCH_F} ⇒ {CRE_MED}	0.001879	1.0	2.1111
{AVG_LOW,EX_MED,GPR_4} ⇒ {CRE_MED}	0.001879	1.0	2.1111
{AVG_MED,EX_MED,GPR_4} ⇒ {CRE_MED}	0.001879	1.0	2.1111
{AVG_LOW,EX_MED,SCH_G} ⇒ {CRE_MED}	0.003759	1.0	2.1111
{AVG_LOW,GPR_4,SCH_F} ⇒ {CRE_MED}	0.001879	1.0	2.1111
{AVG_LOW,SCH_C,EX_LOW,GPR_3} ⇒ {CRE_MED}	0.001879	1.0	2.1111
{AVG_MED,EX_MED,GPR_2,SCH_G} ⇒ {CRE_MED}	0.001879	1.0	2.1111
{AVG_MED,SCH_D} ⇒ {CRE_MED}	0.011278	0.85714	1.8095
{AVG_LOW,GPR_2,SCH_F} ⇒ {CRE_MED}	0.007518	0.8	1.6888
{EX_LOW,GPR_3,SCH_H} ⇒ {CRE_MED}	0.005639	0.75	1.58333
{AVG_MED,SCH_B,GPR_3} ⇒ {CRE_MED}	0.005639	0.75	1.5833
{AVG_HIGH,SCH_A,EX_LOW,GPR_3} ⇒ {CRE_MED}	0.005639	0.75	1.5833
{SCH_D,EX_LOW} ⇒ {CRE_MED}	0.009398	0.71428	1.5079
{AVG_MED,EX_LOW,SCH_H} ⇒ {CRE_MED}	0.009398	0.71428	1.5079
{AVG_MED,GPR_2,SCH_I} ⇒ {CRE_MED}	0.009398	0.71428	1.5079

Table 12. Rules obtained for low performance measured with amount of credits.

Rule	Support	Confidence	Lift
{SCH_B,GPR_4} ⇒ {CRE_LOW}	0.001879	1.0	6.5679
{AVG_LOW,SCH_C,EX_MED} ⇒ {CRE_LOW}	0.001879	1.0	6.5679
{SCH_C,EX_MED,GPR_3} ⇒ {CRE_LOW}	0.001879	1.0	6.5679
{AVG_LOW,GPR_4,SCH_G} ⇒ {CRE_LOW}	0.001879	1.0	6.5679
{AVG_HIGH,EX_LOW,GPR_2,SCH_H} ⇒ {CRE_LOW}	0.001879	1.0	6.5679

values (EX_LOW and EX_MED). About ages, it's not clear as the rules obtained for performance considering GPA, but also can be seen that older groups (GPR_3 and GPR_4) tends to have a lower performance than the most common entrance age (GPR_2) which it's related with the biggest amount of accumulated credits.

Reviewing the rules related with low performance, Table 12, as in the performance measured using GPA, also appears the groups of older ages (GPR_3 and GPR_4). More rules contains the lower average in the former level (AVG_LOW) and not a high mark in the admission exam, appearing only medium and low marks (EX_MED and EX_LOW).

5 Conclusions

Goal of this work was to analyze if the way that UAM is choosing their students, only the mark in the admission exam, allows choosing those ones which will have a good performance. It's evident that a high mark in the admission exam is important, however, it's also clear that for a high performance, considering both GPA and accumulated credits can be useful to consider the former level average, not only as a prerequisite, but as a criteria that can be included in some way, maybe as a percentage, in the admission process.

It's also clear that entrance age plays an important factor over the performance of students, groups of ages greater than the common entrance age tends to have a lower performance, specially the Group 4 (greater than 25 years old). This has a good reason, at that age, it's common that a student has a job and maybe a family, so the time for investing in its studies is less than in a young person which possibly has the support of a relative. UAM gives opportunity to all people for trying to enter, so this criteria cannot be considered as a way for choosing students.

Because of confidentiality reasons, names of origin schools were not included, however, can be mentioned that they don't have a relevant impact over the performance. Besides, as institution, UAM cannot reject a student due to the school it comes, for this reason, this criteria, as the entrance age, can't be considered for choosing students.

Future works consider applying this technique not only for the first year, but for all the time students take for finishing their career. This involves including more variables, because in four or five years exist a lot of situations that can occur, like having a family, a sickness or getting a job while it's less common that this situations happen in the first year. Also, a similar study classifying students according their performance considering factors like average or time for finishing their studies can be interesting for university. This process can be applied not only for analyzing Computer Engineering students, but for all Engineering at UAM, other study areas like Social or Biologycal Sciences and Art and Design Studies can be of interest for UAM authorities. Also, this work can be applied to even other universities.

Acknowledgments. Thanks to Academic Secretary and Scholar System of UAM for providing the data used in this work.

References

1. Azevedo, A. I. R., Santos M. F.: KDD, semma and CRISP-DM: A parallel overview. IADS-DM, 182–185 (2008)
2. Romero C., Ventura S.: Educational data mining: a review of the state of the art. IEEE Transactions on Systems, Man, and Cybernetycs, Part C (Applications and Reviews 6, 601–608 (2010)
3. MunjarD., Keček D., Matotek D.: Relationship between enrolment criteria and first-year students' study-success. In: 3rd Human and Social Sciences at the Common Conference-HASSACC, pp. 111–115 (2015)
4. Superby J. F., Vandamme J. P., Meskens N.: Determination of factors influencing the achievement of the first-year university students using data mining methods. Workshop on Educational Data Mining (2006)

5. Sheikh L. M., Tanveer B., HamdaniM. A.: Interesting measures for mining association rules. In: Proceedings of INMIC 2004 8th International, pp. 641–644 (2004)
6. Maimon O., Rokack L.:Data Mining and Knowledge Discovery Handbook. Springer Science+Business Media (2010)
7. Universidad Autónoma Metropolitana, División de Ciencias Básicas e Ingeniería, http://cbi.azc.uam.mx/es/CBI/Planes_Programa_Estudio_Com
8. Kowitlawakul Y., Brenkus R., Dugan N.:Predictors for success for first semester, second-degree Bachelor of Science in Nursing students. International journal of nursing practice 19, 38–43 (2013)
9. Zebdewos Z., Nasser A., FissehaM.: Predictors of academic achievement for first-year students. The case of Wolaita-Soddo University Ethiopia study-success enrolment. European Scientific Journal 28, 160–173 (2015)

ISSN 1870-4069

Temporal Language Analysis in News Media and Social Networks

Fernando S. Peregrino, David Tomás, Fernando Llopis

University of Alicante, Department of Software and Computing Systems, Spain

{fsperegrino,dtomas,llopis}@dlsi.ua.es

Abstract. The amount of text we can find on the Internet is constantly growing, what makes not feasible to manually analyse such as quantity of information. The Natural Language Processing (NLP) research field has provided a set of tools and techniques that allow human beings to extract relevant data from unstructured pieces of text that come from electronic sources such as digital newspapers or online social networks. The aim of this article is to make a temporal analysis of both formal (newspaper articles) and informal (Twitter messages) texts sources. In this article, we will analyse how some terms evolve in time and the correlation between the formal and informal corpora.

Keywords. Geographical focus detection, geographical information retrieval, natural language processing.

1 Introduction

Nowadays information society is characterised by having a huge amount of information and for the need to allow both a rapid access and disclosure of this information. To this end, storing such information in a digital format has become crucial. Access to this digital information has to be as fast and accurate as possible. With this motivation arise Information Retrieval (IR) systems, also known as search engines.

IR is the science of searching for information in digital documents, producing as a result a subset of the initial documents sorted according to the relevance of a given query. IR deals with the representation, storage, organization of and access to information items such as documents and web pages. The representation and organization of the information items should provide the user with easy access to the information she is interested in [1].

The core tasks of these systems is generally split in two stages: indexing and searching. In the former, all the terms in the documents are indexed. To this purpose, the system records in which documents and how often appears each of these terms, as well as different statistics related to the number terms in every single documents. In the latter, a query is sent to the system to retrieve a set of document sorted by the relevance to this query, trying to match the terms which are in the query with the ones that have been indexed in the previous phase.

Among all the data that IR systems have to deal with, geographical information is a specially relevant type. In accordance with the study conducted by [6], 12.7% over four

pp. 125–136; rec. 2017-09-10; acc. 2017-10-20

million sample queries have a toponym[1] at least. This has been corroborated by the work carried out in [4], where 36 million queries submitted to an IR system were analysed. It was found that between 18% and 22% of these queries where geographically bounded.

Geographical Information Retrieval (GIR) is a specialisation of IR systems, where documents have associated geographical metadata. GIR systems require semantic information, i.e. geographical features associated with the documents. Because of this, in GIR systems document processing and indexing is usually separated from geographical indexing. In other words, each document is indexed according to the place or places in which the document is focused on.

In order to discover weaknesses and opportunities when this geographical index is made, this work conducts a study on different text sources, comparing them to analyse aspects such as the evolution of the use of terms over time and the correlation between formal and informal sources taking into account its geographical scope.

The structure of this article is as follows: Section 2 describes the corpora used in this work; Section 3, provides a description of the analysis and results obtained; finally, Section 4 presents the main conclusions that can be drawn from the work carried out.

2 Corpora

This section describes the corpora analysed in this study. One of the main problems that GIR systems have to deal with is the different nature of texts. If we classify the texts according to their level of formality, they could be divided in two groups:

1. **Formal**. A great amount of formal texts can be found in digital format: newspaper articles, encyclopedia articles, reports, reviews, etc. In this paper we will focus on newspaper articles. More precisely, in the articles belonging to the *20Minutos*[2] newspaper.
2. **Informal**. Online Social Networks have facilitated the existence of huge amounts of texts written in informal language. In this paper we will worked on *Twitter*[3] as a source of this type of texts.

In the two following subsections, we will describe in depth the aforementioned corpora.

2.1 20Minutos

20Minutos is a free Spanish newspaper, with local editions in several Spanish cities. News articles are geographically classified according to the region that they belong to. In this work, the locations that we have worked with have been the 50 Spanish province capitals plus the two Spanish autonomous cities of Ceuta and Melilla. We chose the city capitals since they usually are the most populated cities en their province as well as the administrative location, which makes them a source of a large number of articles

[1] Toponymy is the study of place names (toponyms), their origins, meanings, use, and typology.

[2] `http://www.20minutos.es/`

[3] `https://twitter.com/`

that represent not only the given city but its province. The period of time comprised was from January 1st 2008 to December 31st 2011, inclusive. For all these four years and geographical regions, we obtained a total number of 519,563 geographically tagged newspaper articles.

In order to obtain the articles, we built a crawler which iterated over the section where the local news where published. Each piece of news found from any of aforementioned cities and time period were stored.

A pre-process was carried out on the corpus, removing all the *URLs*, punctuation symbols and special characters such as underscores, slashes, etc. In addition, as part of this pre-process, all the terms were lower-cased, except for the first character of each term, which was kept with the purpose of identifying proper nouns such as locations, people or organisations.

Because of the comparison between *20Minutos* and Twitter corpus in the experiment conducted in Section 3.1, a supplementary set of newspaper articles belonging to the year 2013 were crawled. The dates encompassed in this set was from March 1st 2013 to July 31st 2013. This set of newspaper articles was pre-processed as described in the previous paragraph.

2.2 Twitter

Twitter is an online social networking service that enables users to send and read short 140-character messages called *tweets*. Twitter was created in March 2006 and launched in July 2006. The service rapidly gained worldwide popularity. As of August 2017, Twitter has more than 325 million monthly active users[4].

To obtain our corpus of tweets, we used the Twitter *SEARCH API*, currently included with some restrictions in the Twitter *REST API v1.1*[5]. Through this API, a set of geo-referenced tweets was acquired from the 50 Spanish capital cities plus its two autonomous ones, as in the *20Minutos* corpus (see Section 2.1). The dates of these tweets span from April 20th 2013 to June 10th 2013, 52 days.

Given that Twitter users can send tweets from more than one location in the collected corpus, tweets were grouped by user and location. That is, were we to have a set of tweets from a user which has tweeted from both the location *A* and *B*, those set of tweets are treated as two different sets instead one. The reason for doing this is that when users tweet from a location, in many cases they refer to a place or entity in the place where they are writing, what means it could be useful when it comes to extract features from the given place.

The most relevant data extracted from the obtained tweets is:

- The text of the tweet. Up to 140 characters.
- The tweet location. It is one of the 52 location aforementioned.
- The user. The user who has sent the given tweet.

Notice that if a user has tweeted from n different locations, it will count as n users. What underlies this is that we are trying to figured out where a user is analysing a set

[4] `https://about.twitter.com/company`

[5] `https://dev.twitter.com/docs/using-search.`

of tweets that he or she has sent, and since these set of tweets could differ considerably depending on the location where they were sent, we are splitting these tweet according to the origin, in order to avoid extracting noisy features from the different places.

As in the *20Minutos* corpus, a pre-process was carried out in the corpus of tweets. All the punctuation signs, special characters (underscores, slashes, etc.) and *URLs* were removed. The terms which started with # or @ were kept, since these characters have a special meaning in Twitter, representing *hashtags* and user names respectively, what could be useful to determine the users location.

All the repeated tweets as well as the re-tweeted ones were also removed, as they do not give any additional information. No additional information (e.g. *followers* or location field in the profile) was taken into account.

3 Analysis

This section describes an analysis over the two corpora mentioned in the previous section. This analysis exposes the temporal texts terms evolution according to their geographical and temporal arrangement.

The aim of this analysis is to give a deeper insight into the existing relationship among the used terminology in order to improve the geographical focus detection in GIR systems.

To this end, the following aspect will be analysed in both corpora:

1. Correlation between terms from both corpora.
2. Terms evolution over time.

3.1 Corpora Correlation

In this section, we will show the correlation between the terms of the *20Minutos* and Twitter corpus. To accomplish this, we will measure how similar the texts from both corpora are. This similarity will be carried out comparing the Twitter corpus with a corpus of newspaper articles that encompass three different periods in the year 2013, as described in section 2.1:

1. **Before**. The 52 previous days to the Twitter corpus.
2. **During**. The 52 days which coincide with the Twitter corpus.
3. **After**. The 52 days after the Twitter corpus.

The reason under this corpus split is to corroborate the following hypothesis: the messages expressed in Twitter usually reflect what is happening in day-to-day, what means that a strongest correlation with contemporary newspaper articles must exist. Thus, the terms from each of these three newspaper corpora were compared with the Twitter corpus. This comparison was carried out at city level, i.e., for a given location and period of time (*before*, *during* or *after*) its terms were compared with the terms of the same city in the Twitter corpus.

In order to check the correlation between each of these texts Kullback-Leibler (*KL*) divergence was used [5]. In probability theory and information theory, the KL divergence is a measure of the difference between two probability distributions *P* and *Q*. In practice, *P* represents the real distribution of data, observations, or a precisely calculated theoretical distribution, while *Q* represents a theory, model, description, or approximation of *P*. In other words, it is the amount of information lost when *Q* is used to approximate P.

For discrete probability distributions *P* and *Q*, whose vocabulary is in the set of finite terms χ, the KL divergence from *Q* to *P* is defined by equation 1:

$$D(P||Q) = \sum_{x \epsilon \chi} P(x) log \frac{P(x)}{Q(x)}. \tag{1}$$

Given that the KL divergence is not symmetric, we have calculate the KL symmetric divergence, i.e. the KL distance, based on the approach exposed in [3], as the equation 2 shows:

$$D(P||Q) = \sum_{x \epsilon \chi} (P(x) - Q(x)) log \frac{P(x)}{Q(x)}. \tag{2}$$

In this way, all the terms in the corpus of tweets for each location is obtained along with their frequency. On the other hand, the same operation is carried out with the set of newspaper articles which are included in the period that we want to measure the KL distance with.

The union of all the term from both corpora, that is the vocabulary from both corpora, is expressed by the symbol χ in the equation 2. *P* and *Q* represent the Twitter and *20Minutos* terms frequency normalised vectors respectively.

So as to accomplish this normalisation, the total number of terms and their frequency that appear in each of the given locations is considered. Equation 5 shows the smoothed applied to the terms that are not in the Twitter corpus but exist in χ.

Once all the terms from the Twitter corpus of a given location are normalised, a vector with the weight of all the terms that are in χ is created. For all those terms which are in the vocabulary (χ) but not in the corpus of tweet of the analysed location, a residual value is assigned. This value is calculate dividing the smoothing value ϵ, obtained in the equation 5, by the total number of terms that are in χ but not in the vocabulary of the location under analysis. The frequency of these terms is also considered. Consequently, this normalisation is carry out as is shown in equation 3:

$$P(t_i, c_j) = \begin{cases} \frac{freq(t_i)}{d}, t_i \; is \; in \; the \; corpus \\ \frac{\epsilon}{d}, t_i \; is \; not \; in \; the \; corpus. \end{cases} \tag{3}$$

In this equation, $P(t_i, c_j)$ represents the term *i* probability from the location *j*. $freq(t_i)$ is the stands for the term *i* frequency in the location *j*. *d* is the divisor that distributes the ϵ weight over the terms that are not in the corpus. Finally, ϵ is the smoothing factor applied to the terms which are not in the corpus.

The resultant vector is denoted in the equation 2 by the symbol *P*, whilst *Q* represents the vectors of terms gathered from the location analysed in the *20Minutos* corpus in the given time span. This vector has been built following the procedure employed for vector *P*.

The calculation of the divisor d shown in the definition 3 is achieved in accordance with what is exposed in equation 4:

$$d = np * \epsilon + fp, \tag{4}$$

where *np* represents the number of terms in χ that are not in the analysed corpus, ϵ is the smoothing value, and *fp* is the total number of terms and their frequency which appear in the analysed corpus.

The calculation of the smoothing value ϵ is effectuated as shown in equation 5:

$$\epsilon = \frac{\frac{1}{1+(|vp-vq|)}}{1+p+q}. \tag{5}$$

This equation is based on the work by [3], where $|vp - vq|$ represents the total number of terms which are not shared by the two analysed corpora. p is the total number of terms for the Twitter analysed location, where each term is multiply by its frequency in the given corpus. q stands for the *20Minutos* total number of terms from the analysed location and period multiply by their frequency.

In such manner, KL distance has been calculated at city-level between the Twitter corpus of these locations and these same locations from the *20Minutos* corpus for each of its three periods.

Notice that the smaller the KL distance, the bigger the correlation between corpora. If we look at the 10 largest Spanish locations, in 9 out of 10 (90% precision) the distance is lower in the *During* period, showing a largest correlation between the terms in both corpora when day occur in the same time span.

It is noteworthy the different nature of these text sources, where the *20Minutos* texts are related to current issues in a concrete geographical area in a specific date. On the other hand, in Twitter, we can find a huge range of topics. Frequently, this topics are not related to the location where they were sent and, therefore are dealing with a completely different subject. Actually, in many occasions, these tweets talk about subjects specific from other locations.

We should also keep in mind that there is little geographical information in the vast majority of tweets, which makes more difficult to match these messages with the given local news.

Another aspect to consider is that the three (*before*, *during* and *after*) *20Minutos* corpora used in this analysis where published in consecutive dates, that is these newspaper articles where very close in time, so that the treated issues in these periods overlap each other, specially when it comes to the period in the middle, the one which is supposed to get the minimum KL distance with respect the Twitter corpus.

All in all, having into consideration all the difficulties previously described, we can conclude that our results get a significant accuracy, mainly when it comes to the largest cities.

3.2 Temporal Language Evolution

In this section, we will show the temporal evolution of a set of terms included in both corpora at location granularity. In this way, we will be able to see how the relevance of the selected terms is changing in each of the presented locations, allowing to detect or predict the main citizens' concerns for determined periods of time.

So as to analyse the evolution of this set of terms, corpora were split in periods, as it will be explained in sections 3.2 and 3.2. The terms and their frequency were obtained for each of this periods in each of the corpus locations. With these terms and their frequency, the standard score (*z-score*) [2] of each term was calculate. To calculate this measure, each term in a period was compared with the rest of periods for each given location.

The standard score is the signed number of standard deviations an observation is above the mean. A positive standard score indicates an observation above the mean. The calculation of the standard score is shown in equation 6:

$$z = \frac{x - \bar{a}}{\sigma}, \tag{6}$$

where *x* is the analysed term normalised frequency for a given period of time. $\bar{a}$ is the mean of population, i.e. this same term normalised mean for a given location in all the corpus periods of time. Finally, σ represents the standard deviation of those values.

The set of terms to be analysed was chosen from the list of the main concerns for the Spanish citizens. This list was acquired from the Spanish *Centre for Sociological Research* (CIS: Centro de Investigaciones Sociológicas).[6]

The *CIS* surveys the Spaniards citizens monthly so as to detect their main concerns. For this survey, the *CIS* asks to the survey respondents for choosing their three main concerns from a set of options. The options list can be seen in the *CIS* web page[7]. The *CIS* will finally show the percentage of people that have selected each of these options.

For this analysis, the three chosen subjects are: *paro* (unemployment), *corrupción* (corruption) and *educación* (education). These subjects were among the most important Spaniards concerns according to *CIS* surveys.

In order to capture the three aforementioned subjects in *20Minutos* and Twitter texts, the presence of these terms without taken into account capitalisation or accents was observed. The temporal evolution of these concerns for a set of sample locations (Alicante, Madrid and Sevilla) and the country average (España - Spain) is shown in the sections 3.2 and 3.2. The results of the *CIS* survey is also displayed.

So as to check whether a concern gain or lose importance in each of the selected periods of time for a given location, all the terms that were in the texts of each of the corpora locations in each period were sorted according to their standard score in descending order, i.e., the greater their standard score, the higher in the returned list. Once obtained this sorted list of terms, the position in which the terms appears in the

[6] The *CIS* is a Spanish public research institute for sociological issues `http://www.cis.es/cis/opencms/EN/index.html`.

[7] `http://www.cis.es/opencms/-Archivos/Indicadores/documentos_html/TresProblemas.html`

list and the total number of terms that was in the period of time and location analysed were considered in order to obtain the final score as shown in equation 7:

$$ConcernIndex = 100 - 100 \times \frac{TermPosition}{TotalNumberofTerms}. \tag{7}$$

This equation provides values between 0 and 100, allowing to directly compare them with the values obtained in the *CIS* surveys. Notice that the closer these values are to 100, the more significant concern was shown in that period of time. Once the concern index for a given term in a period of time and location was calculated, it was compared with the rest of periods of time for this same term and location.

20Minutos The *20Minutos* corpus was divided by years, given as a result four different corpus for each location: 2008, 2009, 2010 and 2011.

The obtained results for the main citizens concerns in each of the aforementioned locations are displayed together with the country average and what the *CIS* survey obtained.

***Corruption* temporal evolution**

The evolution of the term *corrupción* in both the *20Minutos* articles and the *CIS* surveys is shown in the figure 1.

In the figure 1, it can be observed how the results between cities and *CIS* clearly differ.

Cities such as Alicante or Seville obtain a similar results than the country average, whilst Madrid has an opposed trend. At a city granularity, the citizens corruption perception can suffer great variations depending on the cases in the given places and when these have happened.

Thus, if a city such as Alicante is analysed, a case like Gürtel[8] that erupted in the Valencia Community at the end of 2007 causes that in the following years the corruption perception was rising, reaching its peak when the former president of this community, Francisco Camps, resigned in the mid of 2011.

If we reduce the geographical focus to Alicante, cases such as Brugal[9] or Caja de Ahorros del Mediterráneo (*CAM: Mediterranean Savings Bank*)[10] increased the corruption perception in 2010 and 2011.

Seville had a similar situation (ERE [11]), what makes a similar perception for this subject in this city.

But for Madrid, the rest of cities displayed and the country average coincide with the trend shows in the *CIS* surveys. This mean that the media is reflecting what was happening at that moment.

Another thing to take into consideration is that, according to *CIS*, corruption did not seem to be among the main concerns for the Spanish people in the analysed years. In the subsequent years, this perception suffered an exponential increase in the *CIS* surveys,

[8] https://en.wikipedia.org/wiki/G\%C3\%BCrtel_case

[9] https://es.wikipedia.org/wiki/Caso_Brugal

[10] https://en.wikipedia.org/wiki/Caja_de_Ahorros_del_Mediterr\%C3\%A1neo

[11] https://es.wikipedia.org/wiki/Caso_ERE_en_Andaluc\%C3\%ADa

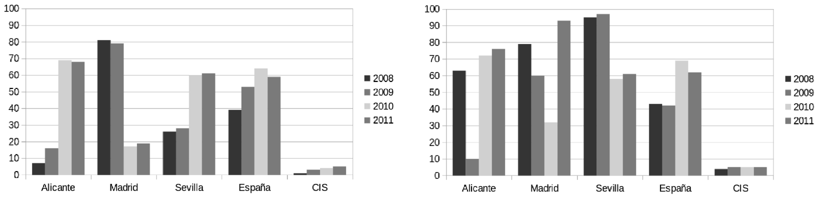


Fig. 1. Left. Corruption (left) and Right Education (right) temporal evolution according to the index of concern of the term *corrupción* in the *20Minutos* articles in the aforementioned locations.

reaching values over 60% in 2014, and around 50% in 2016. We should analyse whether this increase is due to corruption increase or to its pervasive nature in news media, which clearly affects citizens opinions. If we accept the latter reasoning, this would mean that we are able to predict the future citizens concerns through news media avoiding traditional surveys.

***Education* temporal evolution**

The evolution of the term *educación* in both the *20Minutos* articles and the *CIS* surveys is shown in the figure 1.

One more time, belonging each of the location analysed to different communities, and having different policies on education each of these communities because they have derived this competency, it can be observed large variations between locations. This variations depend on the measures and funding cuts adopted in each of their respective communities. Still, it can be seen as the concern for education is growing nationally in the journal texts as the economic crisis progresses, and therefore cuts. Finally, among the cities shown, a similar degree of concern is displayed, which is far above from the one shown in the *CIS* surveys.

According to the *CIS*, the degree of concern by the citizenship for education in the years studied was around 5%. This degree has grown steadily, doubling the initial figure in 2016. Again, if you look at the rate at the national level, it appears that what is shown in the press anticipates what would later collect *CIS* surveys. This may also be because the measures taken by the government, generally cuts in the middle of the financial crisis, which echoes the press, have not a final impact until later. After this time, it is when the public begins to notice the effect of the measures taken, and therefore to express concern.

***Unemployment* temporal evolution**

The evolution of the term *paro* in both the *20Minutos* articles and the *CIS* surveys is shown in the figure 2.

For the 4 years analysed, it should be noted that unemployment was the biggest concern among Spanish citizens based on *CIS* surveys. This time, the concern shown by the *CIS* matches that shown in *20Minutos* in the given cities and the country's average.

According to the *CIS*, unemployment concern was growing during these 4 years analysed. Something like the example of Alicante, which is the most similar to those listed in the chart to the national average, with the exception of the last year (2011) that

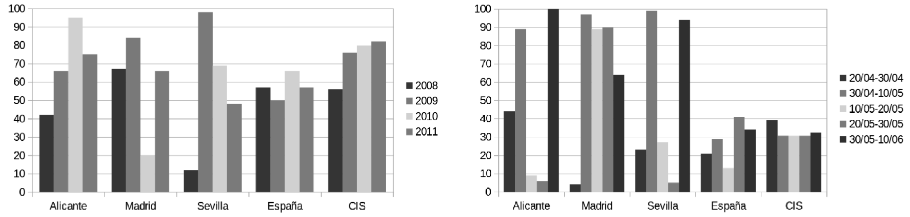


Fig. 2. Left.Unemployment temporal evolution according to the index of concern of the term *paro* in the *20Minutos* articles in the aforementioned locations.Right Corruption temporal evolution according to the index of concern of the term *corrupción* in the *Twitter* messages for the aforementioned locations.

began to decline. According to *CIS*, although unemployment remains the main concern of the Spanish people in 2016, it has decreased to levels that are between those shown for the years 2009 and 2010, which already could be observed on the results of the newspaper *20Minutos* nationwide.

Beyond that, the results shown by the media may affect the opinion of citizens, according to our study, these media seem to be a barometer of what would citizens opine in the future.

Twitter The *Twitter* corpus was composed by tweets collected in 52 different days. In order to be able to observe a temporal evolution of terms we decided to divide the corpus into five parts each of these divisions comprising periods of 10 days (the first and last period have 10 days plus the initial or final day respectively): 20/04/2013-30/04/2013, 01/05/2013-10/05/2013, 11/05/2013-20/05/2013, 21/05/2013-30/05/2013 y 31/05/2013-10/06/2013.

In the following sections, the evolution of the concerns, which were previously mentioned in the sample locations indicated in the figures, will be shown.

This time, it should be noted that since the *CIS* shows the results of their survey monthly, and because of each analysed period covers only 10 days, some of the *CIS* surveys values shown in the figures (the ones that comprise April) do not vary in several periods.

It also should be noted that because the time periods cover only 10 days, the results obtained from the standard score may considerably fluctuate.

***Corruption* temporal evolution**

The evolution in the presence of the term *corrupción* in *Twitter* between the indicated dates is displayed in the figure 2. This concern index is compared to that shown in surveys conducted by the *CIS* for this same time period.

In social networks, the average rate of concern for the term *corrupción* for the period of time indicated has been lower than the one shown in the newspaper *20Minutos* for the exposed years. Nevertheless, Spanish people concern about this subject has considerably increased (around 30%) as it can be observed in the figures 1 and 2.

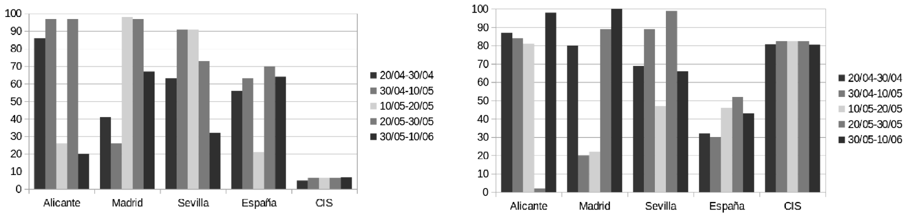


Fig. 3. Left. Education temporal evolution according to the index of concern of the term *educación* in the *Twitter* messages for the aforementioned locations. Right Unemployment temporal evolution according to the index of concern of the term *paro* in the *Twitter* messages for the aforementioned locations.

Among the displayed cities, it is visible how Alicante and Seville have a virtually identical trend for the term *corrupción*. On the other hand, this concern did not decrease in Madrid in the third and fourth period as it did for the other two locations.

We must highlight how the average score for all the cities is very similar to the achieved scores in the *CIS* surveys, especially when we calculate the average for the month of April, such as is made in the *CIS* survey. Thus, *Twitter* seems to accurately reflect what the *CIS* surveys show.

***Education* temporal evolution**

The evolution in the presence of the term *educación* in *Twitter* between the indicated dates is displayed in the figure 3. This concern index is compared to that shown in surveys conducted by the *CIS* for this same period of time.

This time, the term *educación* has obtained more differentiated values than with the previous term. The fluctuation experimented in the concern rate of this subject in *Twitter* has been significantly high, varying between 20 and 100 percentage points. This is due to the different meanings of the term '*educación*' in Spanish. The main senses of this term are education and good manners, more than academic education.

According to the graph shown in the figure 3, Alicante seems to be a meaningful sample that let us know how *education* has been perceived in the rest of the country in *Twitter*.

On the other hand, in accordance with the data shown in *CIS* and *Twitter*, concern for education is much higher in the social network than in the published survey. In *Twitter* do not seem to be one of the main concerns of citizens, as happened with that shown in the *CIS* results in Figure 3, where the results of the *CIS* surveys were compare with the results in the *20Minutos* newspaper. Despite this, comparing the *CIS* results from both graphs, it is indeed perceived to have a considerably increased from one date to another.

***Unemployment* temporal evolution**

The evolution in the presence of the term *paro* in *Twitter* between the indicated dates is displayed in the figure 3. This concern index is compared to that shown in surveys conducted by the *CIS* for this same time period.

Regarding the term *paro*, the data shows an unequal concern rate between the different cities. This time, unemployment is more relevant in the *CIS* surveys (it is clearly the main concern) than the average value obtained from the social network.

On the other hand, as previously mentioned, when such a short periods of time are analysed, great fluctuation can be found in the standard score values. If we omit the fourth time period, we can observe how this concern follows a very similar evolution between *CIS* and *Twitter* for the city of Alicante, as well as Seville if we instead omit the second period of time.

4 Conclusions

In this article, two different studies have been carried out which were intended to provide information on both formal and informal texts, the corpus of *20Minutes* and *Twitter* respectively.

In the experiment carried out to show the correlation between these corpora, the results show that there is a temporal correlation that reaches 90% in the top 10 largest cities of the country. This cities are in turn the ones from which a great number of tweets have been collected in the corpora.

This is a great result considering the difficulties of the task, such as: the different nature of the corpora, messages sent in *Twitter* usually refer to other places, little geographic information found in the tweets, or that the dates of the articles in the newspaper corpus were contiguous.

As for the experiments that showed the temporal evolution of the language, as it has been observed in the results obtained, it seems that what the press publishes, after a while, is reflected in the polls of the *CIS*. This can be useful to predict trends in the concerns that citizens will face in the future, either because the press anticipates these concerns or because it influences citizens.

Apart from that the results shown by news media can affect the opinion of the citizens, according to this study, these media seem to be a thermometer of what citizens will be concerned in the future.

With regard to *Twitter* and the temporal evolution of the terms, it would have to be followed in a longer period of time, since if these periods are divided into 10 days large shifts in the evolution of terms take place, especially when considered at city level.

References

1. Baeza-Yates, R., Ribeiro-Neto, B., et al.: Modern information retrieval, vol. 463. ACM press New York (1999)
2. Benzécri, J.P., Bellier, L.: L'analyse des données, vol. 1. Dunod Paris (1976)
3. Bigi, B.: Using Kullback-Leibler distance for text categorization. Springer (2003)
4. Gan, Q., Attenberg, J., Markowetz, A., Suel, T.: Analysis of geographic queries in a search engine log. In: Proceedings of the first international workshop on Location and the web. pp. 49–56. ACM (2008)
5. Kullback, S., Leibler, R.A.: On information and sufficiency. The annals of mathematical statistics pp. 79–86 (1951)
6. Zhang, W.V., Rey, B., Stipp, E., Jones, R.: Geomodification in query rewriting. In: GIR (2006)

ISSN 1870-4069

On Sequence Encodings for Positional Reasoning Task with Deep Neural Networks

Ari Reyes, Hiram Calvo

National Polytechnic Institute, Center for Computing Research,
México

a160848@sagitario.cic.ipn.mx,hcalvo@cic.ipn.mx

http://www.cic.ipn.mx

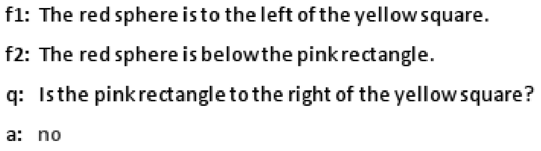
Abstract. This paper addresses the problem of encoding natural language in neural networks for the task of question answering on positional facts. Current state of the art works use many different ways to encode their inputs in natural language. Most of them separate each fact and their interaction with the question is independent. Another common issue is that, when encoding is not based on bags of words, sequence of words is considered, but the effect of alignment has not been particularly studied. In this paper we propose representing the intermediate states of a Recurrent Neural Network (Particularly a Long Short Term Memory network) as a matrix, and then using a convolutional layer on it. This architecture allows to experiment with different strategies of word alignment, as well as different modes of interaction between facts and questions, including a 3D convolution to combine word alignments and interaction of all facts and the question to be answered. We apply this model to the Positional Reasoning Task of bAbI to evaluate our proposed models. We found that alignment does not play a very important role in this task, but allowing interaction between all facts and question simultaneously is important to improve performance.

Keywords. Question answering, LSTM, CNN, deep learning, sequence encodings.

1 Introduction

Question answering (QA) is a complex problem within natural language processing that involves understanding a question and reasoning about provided facts presented in order to give an answer the question. Many efforts have been focused on building rule-based solutions [4]; however, due to the great flexibility and versatility of natural language, coupled with the fragility of these solutions, one of the major challenges for these systems has been the encoding of natural language into a formal language in such a way that it allows finding an answer using rules of inference.

On the other hand, recently different models based on neural networks, such as Memory Network [8], Dynamic Memory Network [3] and Neural Reasoner [4],



f1: The red sphere is to the left of the yellow square.

f2: The red sphere is below the pink rectangle.

q: Is the pink rectangle to the right of the yellow square?

a: no

Fig. 1. Example facts, question and answer from bAbI Task 17: Positional Reasoning.

have been proposed. All of them use deep neural networks with the goal of answering factual questions.

Notwithstanding, the problem of encoding natural language is still a gist for these systems. Every model uses different ways of encoding their inputs. Most of them isolate the interaction of the question to be answered with each fact separately; additionally, when not considering bags of words, sequence of words is taken into account, but, to our knowledge, the effect of alignment has not been particularly studied. Our hypothesis is that, by allowing all the intermediate states of a Recurrent Neural Network (Particularly a Long Short Term Memory network—LSTM) to be represented as a matrix, and then using a convolutional layer on it, the effect of misalignment could be undertaken. The same strategy can be used to allow interaction between question and all facts, when they are represented as a matrix as well. In this paper we present different models in order to cover these issues, and finally we propose using a 3D convolution to allow both kinds of interaction: different alignments along with all facts and the question to be answered.

One of the datasets used to evaluate neural network model for QA is the bAbI tasks dataset [7]. The bAbI tasks were created to measure the progress in the development of an intelligent dialog agent, which allow altogether to evaluate the reading comprehension of a system using Question Answering [7]. Particularly, in this paper we use the task Positional Reasoning—See Figure 1 for an example of the problems from this task—in order to verify our hypothesis.

The rest of this paper is organized as follows. In Section 2 we discuss related works, then in Section 3 we present our model with four variations proposed to answer the aforementioned questions. In Section 4 we give details of our experiments and results, and finally in Section 5 we draw our conclusions and outline some possibilities for future work.

2 Related Work

As far as we know, a study of the impact of using different encodings in the sequence of words to obtain a semantic representation of a sentence for this task (bAbI Task 17: Positional Reasoning) has not been conducted. As described in the previous section, we are interested on determining the convenience of modeling facts and question using the interaction of the words contained in them, and afterwards applying a sequential flattening; or, applying this sequential flattening first and then allowing the interaction of these sequential representations.

In general, works dealing with bAbI tasks are mainly based on neural networks with peculiar modifications to the classical models handled by these algorithms. These works enrich the wide panorama in which these models are developed allowing flexible and versatile solutions for the problems presented in question answering. One of these works introduces Memory Networks [8] whose main contribution is the addition of memory to a RNN.

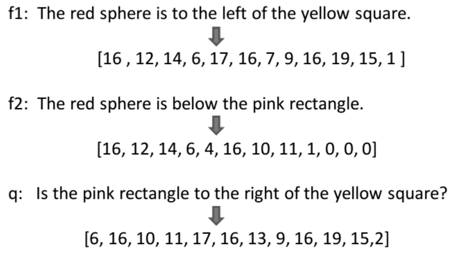
A modification to the previous work is presented in the End-to-End Memory Network [6] model, that can be seen as a kind of recurrent neural network that allows the handling of a memory that only produces an output after a fixed number of time steps with intermediate steps that update the internal state of the memory.

Another work brings in Dynamic Memory Networks (DMM) [3], that use an episodic memory with which it is possible to link the facts that relate more directly to the question leaving aside those that have no relevance to the answer.

A work even closer to the models proposed in this article is presented in [4]. The Neural Reasoner has a first layer that, using recurrent neural networks, encodes the facts and question separately, and then applies several layers of reasoning based on DMNs to obtain a solution to tasks 17 and 19 of the bAbI dataset.

Each one of these works tackles the problem of encoding natural language in different ways. For example, [8] encodes facts in a separate memory that is combined with the stream of words, [7] experiment with adding a special ”stop” class to separate facts and questions, using bags of n-grams instead of bags of words, and a multilinear map, i.e., a linear map of each word depending on its position. Finally, the Neural Reasoner [4] proposes using an encoding layer (based on Recurrent Neural Networks – RNNs) before all reasoning layers. This encoding layer later allows the question to interact separately with each fact through a Deep Neural Network (DNN), and finally, outputs of each DNN are combined to form an answer in the last layer by means of different pooling strategies.

In rough terms, the first two models propose a separate memory, while the last one, permits interaction between facts only after they have been compounded with the question. Several research interrogations arise from these works. For example, given that facts sometimes refer to relations between the same objects, would it not be advantageous to allow all of them to interact with the question, as well as between them? Another issue is related to the influence word position has in the encodings. If we model each fact and the question as sequences of words, then they must be aligned. How much does this affect the final result? And finally, sequential networks change their states as new inputs are considered. If we keep these changes as a matrix, then it would be possible for past states to interact with new states, rendering the problem of alignment as not important. We propose a model with different variations aiming to have a better panorama in the realm of these questions.



f1: The red sphere is to the left of the yellow square.

[16 , 12, 14, 6, 17, 16, 7, 9, 16, 19, 15, 1]

f2: The red sphere is below the pink rectangle.

[16, 12, 14, 6, 4, 16, 10, 11, 1, 0, 0, 0]

q: Is the pink rectangle to the right of the yellow square?

[6, 16, 10, 11, 17, 16, 13, 9, 16, 19, 15,2]

Fig. 2. Example left alignment in the numeric representation.

3 Proposed Model and Variations

In order to find answers to questions posed in the last section, we propose a neural model with different variations, so that we can verify: (1) the effect of allowing all facts and questions to interact before reasoning is carried out; (2) allowing past states of the sequential parts of the model to be stored in a matrix, so that the problem of misalignment is minimized; and (3) incorporating both variations in a single model.

Below is a general overview of the 4 models proposed for the solution of the Positional Reasoning task.

Within this task we have positional facts, see Figure 1, which will be subsequently named $f_1, f_2, ..., f_\mathrm{k}$; and a question, labeled as q.

For all models, a numerical representation is made first for each word from each fact f_i and question q according to a dictionary obtaining vectors of the same dimension n, which is based on the largest number of words found in the facts or questions. For those sentences with a smaller number of words than n, empty spaces are padded with zeros. This alignment can be to the right or to the left. For example, in Figure 2, $n = 12$ and f_2 is aligned to the left.

Subsequently, a vector representation for each word from each fact and question is obtained from the pre-trained 50-dimensional word vectors generated by GloVe [5].

In this way, we obtain matrices of size $n \times 50$ for the representation of each fact, which we will refer now as $F_1, F_2, ..., F_\mathrm{k}$ as well as the question Q.

In all models we use layers of *dropout* in order to avoid the co-adaptation of feature detectors; in other words, to avoid overfitting, and with this, to make the models more robust by reducing the adaptation to noise, following [1].

In the final part of all models there is a layer called Dense. This layer is a completely connected layer to all the vocabulary found in facts, questions and answers whose function is to convert the numerical answer to a word for each given question. See Figures 9 and 12.

3.1 Model 1

For this model we propose a scheme in which matrices obtained for each event $(F_1, F_2, ..., F_\mathrm{k})$ and Q are added. See equation 1:

$$S = Q + F_1 + F_2 + ... + F_k. \tag{1}$$

This resulting matrix S is passed to a recurrent neuronal network, specifically to an LSTM [2], which allows to encode the interaction of words in a sequential form.

Then, the values of each cell of the recurring network are obtained by forming an array with these interactions. Next, a convolutional network is applied to perform an analysis on these sequential representations. This process is shown in detail in Figure 4.

Once Matrix C is obtained, a convolutional network is applied. The purpose of this layer is to blend information of the obtained sequential representations. This layer uses 100 filters, and it is connected to a flattening layer that is connected to all known vocabulary. Then, an answer is selected as shown in Figure 9. This last flattening layer is labeled as *Dense*.

This model's block diagram can be found in Figure 3. Dropout layers were added as they were used in the experiments.

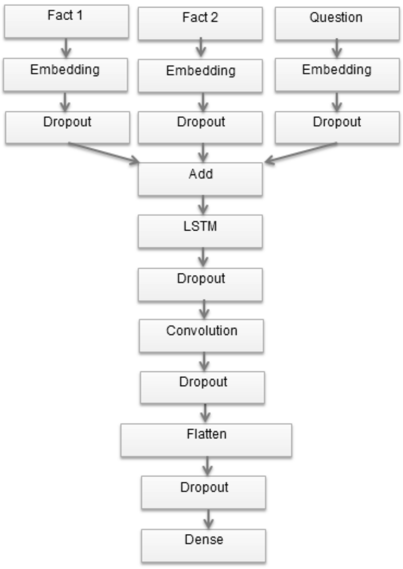


Fig. 3. Block representation of Model 1.

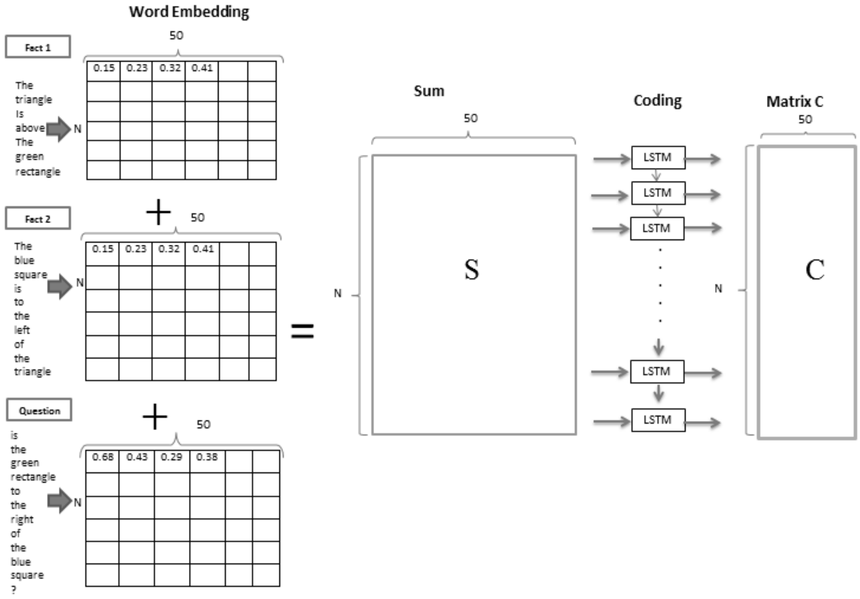

Word Embedding
50
Fact 1
The triangle Is above The green rectangle
N
0.15 0.23 0.32 0.41
+
50
Fact 2
The blue square is to the left of the triangle
N
0.15 0.23 0.32 0.41
+
50
Question
is the green rectangle to the right of the blue square ?
N
0.68 0.43 0.29 0.38
=
Sum
50
N
S
Coding
LSTM
Matrix C
50
N
C

Fig. 4. Obention of Matrix C after LSTM coding, based on the S matrix from Model 1.

3.2 Model 2

Model 2 consists in concatenating instead of adding up matrices. That is, F_1, F_2, ..., F_k and Q are concatenated. This is done by creating a final matrix of $((n \times m)+1) \times 50$, where m represents the number of positional facts and n the largest number of words contained in a fact or question. This process is depicted in Figure 6, showing the way in which matrices are concatenated and then entered into the recurring network LSTM.

In the same way as in Model 1, after the previous step we obtain a matrix C that continues with the same procedure of Model 1 that was described in Figure 9. Model 2 is presented in Figure 5.

3.3 Model 3

For Model 3, each matrix $(F_1, F_2, ..., F_\mathrm{k})$ and Q, is, word by word (in its vectorial representation), passed into a LSTM recurrent neuronal network. This change allows first to codify sentences in a sequential way to obtain separate matrices representing the semantics of each fact and the question. These new matrices are denominated $C_1, C_2, ..., C_k, C_{k+1}$ respectively. Subsequently the matrices obtained by this last step are added to obtain a final matrix C. This process is detailed graphically in Figure 8.

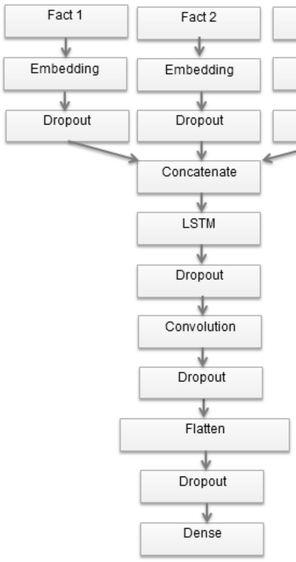


Fig. 5. Block representation of Model 2.

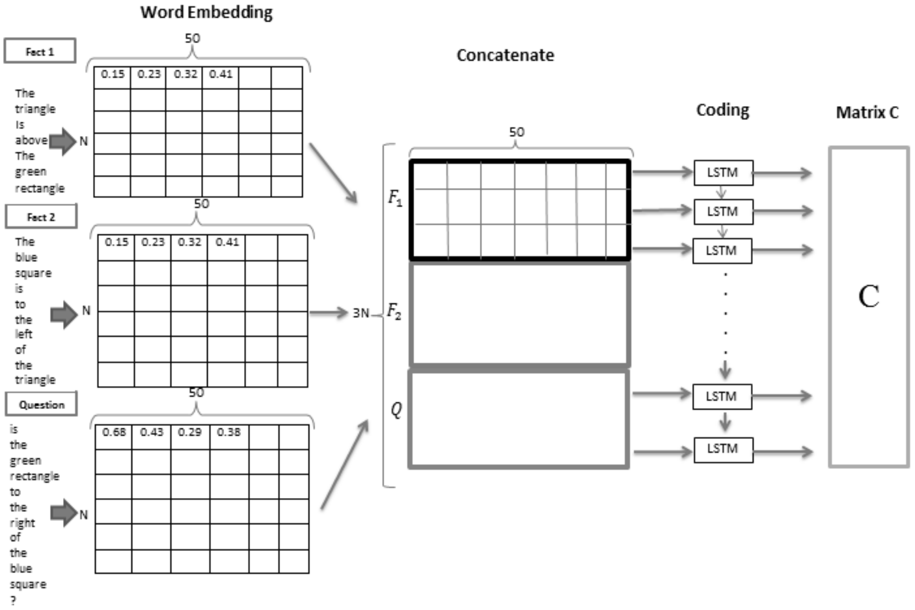


Fig. 6. Concatenation of matrices before LSTM as part of Model 2.

Again, to this point a matrix C is obtained on which a convolution is applied in the same way as in previous models (See Figure 9). Block diagram for this model is presented in Figure 7.

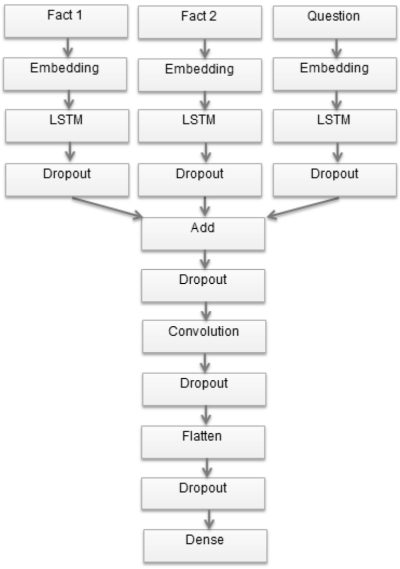


Fig. 7. Block representation of Model 3.

3.4 Model 4

Finally, in Model 4 a modification is made to Model 3 where, instead of adding the matrices obtained by each LSTM, a concatenation of these matrices is done in such a way that a three-dimensional matrix is obtained where each channel contains the sequential representation of a fact and of the question, allowing to make a 3D Convolution on all channels that compose this matrix (see Figure 10).

The intuition behind this model can be seen in Figure 11. This figure shows how matrices obtained from the sequential coding of sentences, now referred to as $C1, C_2, ..., C_k, C_{k+1}$, are concatenated to obtain a representation in a new matrix C.

Obtaining this matrix C, which represents a matrix with 3 channels, allows to do a convolution on these channels. The procedure for this is very similar to the previous models except for the convolution mask, that now resembles a cube that travels through matrix C. This the last process of model 4. See Figure 12.

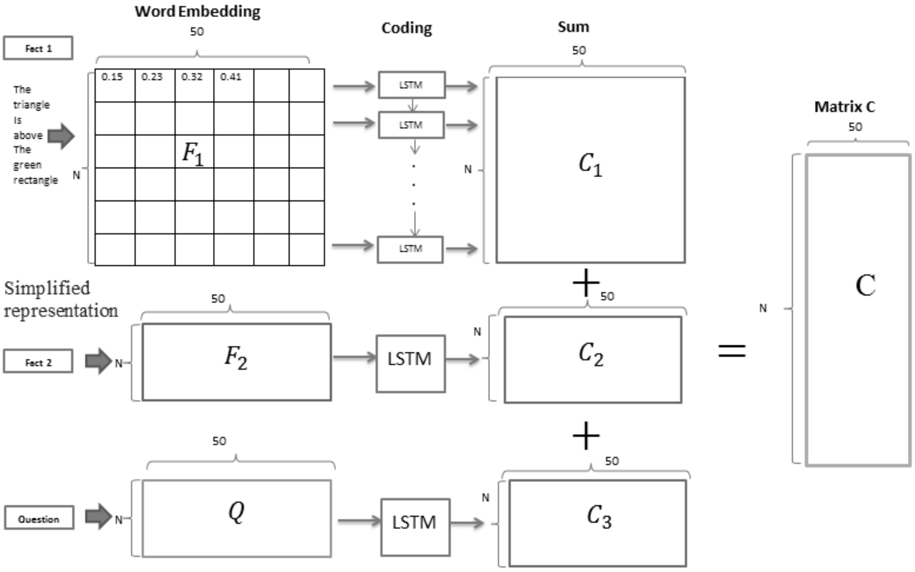


Fig. 8. Addition of matrices from the LSTM output as part of Model 3.

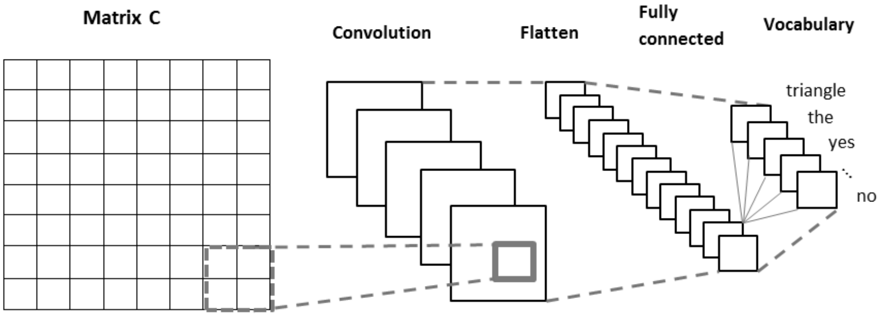


Fig. 9. Part 2 of the detailed Model 1,2 & 3 process.

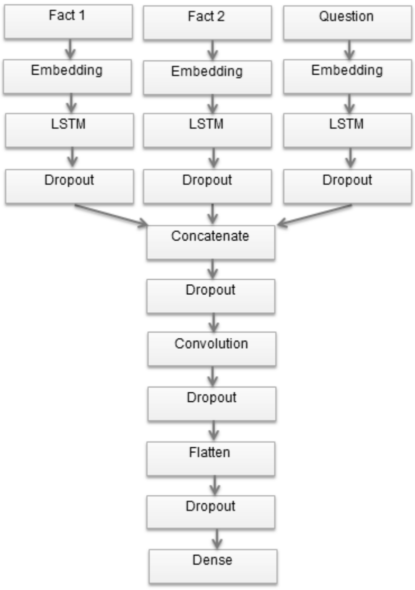


Fig. 10. Block representation of Model 4.

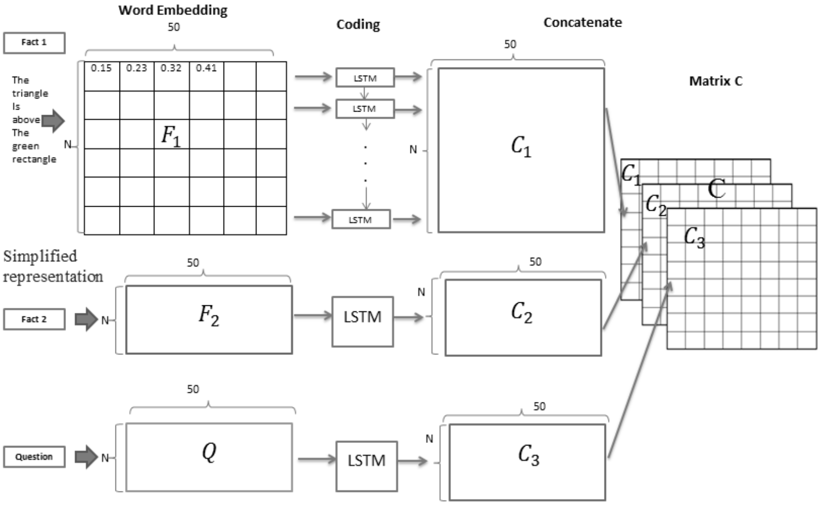


Fig. 11. Concatenation of Matrices in first part of Model 4.

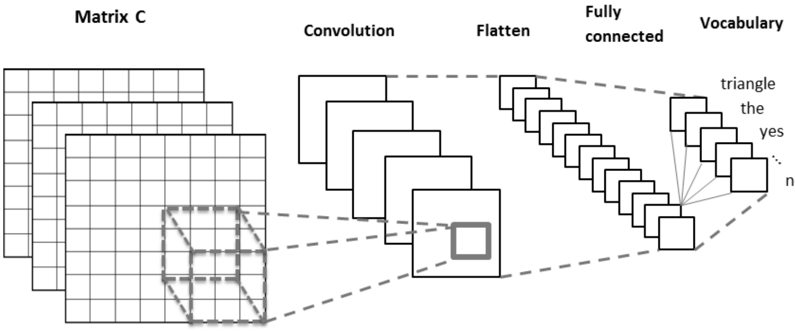


Fig. 12. Three dimensional convolution in Model 4.

4 Experiments and Results

For these experiments, the subset 1k from bAbI Task 17 was used. 1,000 instances were used for the training, and 1,000 instances for testing. 200 epochs and a batch size of 32 were used. Alignment to the right and to the left in the numerical representation of the facts and questions was made to obtain conclusions about the effect of alignment of sentences in each model. For the convolutional networks the ReLu activation function was used and the number of filters used was 100. The experiment was carried out with different mask sizes in the convolution, see Section 4.2. In dropout layers, a value of 0.3 was used.

4.1 Efect of Alignment

We experimented with an alignment to the left (see Figure 2) and alignment to the right (by placing zeros on the opposite side) in the numerical representation of the sentences in order to attest its effect in the models. By this means, we can verify if there is any influence of the sequential coding of sentences using LSTM recurrent neural networks. Results of this experiment can be seen in table 1.

4.2 Variation of the Size of Masks in Convolution

We experimented with varying the size of the mask in the filters in convolutional networks. Particularly, 2x2, 3x3 and 4x4 masks were used allowing different interactions between the sequential representations of the models. The idea behind the variation of these masks is to allow the interaction between a greater or a lower number of words, aiming to obtain better characteristics on the sequential representations of each sentence. For this task, the convolution (also called 3D convolution in Model 4), has a maximum number of channels of 3, because there are at most three channels: one for the first fact, one for the second fact, and a third one for the question.

Table 1. Results on the models using right alignment, left alignment and different kernel sizes.

Model	Kernel size	Alignment	Accuracy on training	Accuracy on test
Model 1	2x2	right	0.782	0.617
Model 1	3x3	right	0.776	0.606
Model 1	4x4	right	0.799	**0.624**
Model 1	2x2	left	0.771	0.619
Model 1	3x3	left	0.766	0.587
Model 1	4x4	left	0.732	0.623
Model 2	2x2	right	0.812	0.558
Model 2	3x3	right	0.78	0.555
Model 2	4x4	right	0.758	0.587
Model 2	2x2	left	0.791	0.559
Model 2	3x3	left	0.785	0.599
Model 2	4x4	left	0.754	0.582
Model 3	2x2	right	0.97	0.579
Model 3	3x3	right	0.952	0.549
Model 3	4x4	right	0.949	0.583
Model 3	2x2	left	0.946	0.579
Model 3	3x3	left	0.933	0.568
Model 3	4x4	left	0.93	0.578
Model 4	2x2x2	right	0.792	0.507
Model 4	3x3x3	right	0.793	0.532
Model 4	4x4x3	right	0.827	0.545
Model 4	2x2x2	left	0.772	0.526
Model 4	3x3x3	left	0.793	0.534
Model 4	4x4x3	left	0.805	0.509

4.3 Comparison with the State of the Art

Results of Table 2 were obtained from related works ([7], [6],[3],[4]), considering the best results achieved by each algorithm. We include the best result obtained by Model 1 within this comparison.

It can be observed that Model 1 is above algorithms like the Dynamic Memory Networks and End-to-End Memory Networks; however its performance is below the Neural Reasoner. We did not consider the version of Neural Reasoner with auxiliary information, because external resources were not used, as with plain Neural Reasoner.

5 Conclusions and Future Work

We found that concatenation of matrices was not useful within the interactions of the sequential representations of the words, since Models 2 and 4 have a lower performance than Models 1 and 3 that add matrices, instead of concatenating them. Because models adding matrices performed better, we can support the

Table 2. Accuracy on the task 17 Positional reasoning for different algorithms.

Algorithm	Positional Reasoning (1K)
N-gram Classifier	46.0%
LSTM	51.0%
MemN2N	59.6%
Dynamic Memory Networks	59.6%
Structured SVM	61.0%
Model 1 (4×4)	*62.4*%
Neural Reasoner	66.4%

part of our hypothesis that blending facts and questions yields better results than restricting them to interact separately.

It was also observed that applying a layer of RNNs before adding or concatenating is not favorable to the results, probably because of the interaction between the linear substructures of the vectors generated from GloVe.

It is also possible to conclude that the effect of the alignment in the sentences is not so important due perhaps to the effect of the LSTM, which allows flexibility in this aspect.

Apparently larger masks in convolution yield better results. Part of our future work is to experiment with new shapes and larger sizes in these masks.

Despite the hypothesis that allowing interaction between all words in facts and questions seemed logical, doing a 3D convolution on the channels composed by matrices of all words from positional facts and the question had the lowest performance, as we attested with results of Model 4.

As future work, a parser can be implemented to obtain another a structured representation of sentences. In addition to this, other models that involve more convolution layers or pooling layers could be implemented in order to obtain better characteristics on the sequential representations.

Additionally, proposed models will be tested in the other bAbI tasks to obtain a better evaluation of the performance of these algorithms.

Acknowledgments Authors wish to thank the support of Mexican Government (SNI, SIP-IPN, COFAA-IPN, and BEIFI-IPN). This work was partially funded by CONACYT under the Thematic Networks program (Language Technologies Thematic Network project 281795).

References

1. Baldi, P., Sadowski, P.: Understanding dropout. In: Burges, C.J.C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 26, pp. 2814–2822. Curran Associates, Inc. (2013), `http://papers.nips.cc/paper/4878-understanding-dropout.pdf`

2. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation 9(8), 1735–1780 (1997)
3. Kumar, A., Irsoy, O., Ondruska, P., Iyyer, M., Bradbury, J., Gulrajani, I., Zhong, V., Paulus, R., Socher, R.: Ask me anything: Dynamic memory networks for natural language processing. In: International Conference on Machine Learning. pp. 1378–1387 (2016)
4. Peng, B., Lu, Z., Li, H., Wong, K.: Towards neural network-based reasoning. CoRR abs/1508.05508 (2015), `http://arxiv.org/abs/1508.05508`
5. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Empirical Methods in Natural Language Processing (EMNLP). pp. 1532–1543 (2014), `http://www.aclweb.org/anthology/D14-1162`
6. Sukhbaatar, S., Szlam, A., Weston, J., Fergus, R.: End-to-end memory networks. In: Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R. (eds.) Advances in Neural Information Processing Systems 28, pp. 2440–2448. Curran Associates, Inc. (2015), `http://papers.nips.cc/paper/5846-end-to-end-memory-networks.pdf`
7. Weston, J., Bordes, A., Chopra, S., Rush, A.M., van Merriënboer, B., Joulin, A., Mikolov, T.: Towards ai-complete question answering: A set of prerequisite toy tasks. arXiv preprint arXiv:1502.05698 (2015)
8. Weston, J., Chopra, S., Bordes, A.: Memory networks. CoRR abs/1410.3916 (2014), `http://arxiv.org/abs/1410.3916`

ISSN 1870-4069

PMIIDAS: Integration of Open Educational Resources Based on Semantic Technologies

Cristal Karina Galindo[1], R. Carolina Medina-Ramírez[1], María Auxilio Medina[2], José Luis García-Cué[3]

[1] Universidad Autónoma Metropolitana, Department of Electric Engineering, Iztapalapa, Mexico

[2] Universidad Politécnica de Puebla, Department of Posgraduate, Mexico

[3] Colegio de Postgraduados, Campus Montecillo, Texcoco, Mexico

cdgalindod@gmail.com, cmed@xanum.uam.mx, mauxmedina@gmail.com, jlgcue@colpos.mx

Abstract. This article presents a prototype called PMIIDAS. The prototype focuses on the creation of open educational resources based on pedagogical and institutional guidelines, their representation using semantic indexing, and the storage and retrieval thereof by Semantic Web technologies. The use of this prototype can have benefits such as the reuse of open educational resources, which are distributed in institutional repositories or educational platforms by applying a semantic approach, search and retrieval of relevant information can be done through the use of ontologies and user profiles; as well as the exchange of educational resources among educational institutions.

Keywords. Applications, education, open educational, semantic web, ontology.

1 Introduction

By the Budapest open access initiatives [1], Bethesda [2] and Berlin [3] worldwide have motivated the creation of technological platforms such as the Institutional Repositories (IR), which are implemented in a wide variety of universities and research centers in order to be able to disseminate, share and transfer scientific and academic production, which is embodied in different educational resources. IRs can be considered as institutional memories preserving and increasing knowledge embodied in educational resources [4].

An IR can be understood as an information system that collects, preserves, disseminates and gives access to the scientific and intellectual production of an educational or scientific institution. In addition to having mechanisms to store, preserve and recover educational resources [5] [6]. These educational resources are described through metadata, which conforms to the standard OAI-PMH (Open Archives Initia-

pp. 151–165; rec. 2017-09-10; acc. 2017-10-20

tive Protocol for Metadata Harvesting) [7], which is implemented as a protocol for interoperability in different IRs such as DSpace [8] E-Prints [9], Opus [10], etc.

On the other hand, the educational resources contained in the IRs when adopting open access policies become open educational resources. The United Nations Organization for Education, Science and Culture [11] defines an Open Educational Resource (OER) as any teaching, learning or research material which is in the public domain or which has been published under an intellectual property license that allows its use, adaptation and free distribution.

The opening of resources refers not only to free access to query them, to enrich them or to generate new resources from the original resources, but also to an open structure (semantic indexing) in a processable format (RDF, Resource Description Framework) [12]. Semantic Web technologies are made up of ontologies, descriptions of resources (semantic indexes) and languages for managing and representing of knowledge. The use of these technologies makes it possible to dynamically and actively reach the user by stimulating the use of applications and achieving a shorter time in the search for open educational resources [13].

This article presents the Prototype PMIIDAS, which is the acronym for Integrating Memory of Documentary Information Sources for Educational Institutions Based on Semantic Web Technologies; such tool can be defined as an organizational memory for educational purposes. The knowledge possessed by people (professors, learners and officers) and by educational resources (which are heterogeneous in content, format and type) is materialized explicitly and persistently in this tool in order to contribute to their reuse and decision making among members of the educational institution [12].

This prototype can be seen as a proposal to concentrate OERs within an educational institution, but also as an integration tool for different educational platforms. This prototype focuses on the creation of OERs based on institutional and educational guidelines, their representation using semantic indexing, and the storage and retrieval thereof by Semantic Web technologies.

The rest of the article is structured as follows: 1) Section II presents a review of the works related to education based on semantic Web technologies, as well as the institutional repositories that have emerged; 2) Section III discusses the methodology and the different stages, as well as the different activities that were used in the proposed work; 3) Section IV shows the process used for prototype design; 4) Section V shows the different tests on the queries made to the ontologies and the tests carried out on the prototype; and finally 5) Section VI shows the conclusion.

2 Related Work

In this section we present the description of some works that have been carried out in education and that use the semantic approach; as well as institutional repositories.

One of the main works in education based on the semantic approach is the ELENA project [14], which is a European Community project whose primary objective is to allow the creation of a “Smart Learning Space” by standards, educational mediators and semantic Web technologies as key factors in the integration of e-learning educational resources.

The article titled “A computational model for developing semantic web-based educational systems” [15] presents a computational model for the development of Web-based Semantic Educational Prototypes, focusing on adaptive learning.

The Didactalia project [16] proposes a personal learning environment including semantic and social tools, having a space in which the user collects show and share learning contents.

Procomún [17] is an intelligent, social and distributed Web, which is part of an educational environment under the building. Such tool can connect with educational communities in existing spaces by the Linked Open Data Cloud [18] using its semantic properties.

The work of Medina et al. [19] proposes a strategy to produce and consume linked open data from the OAI-PMH protocol, which is compatible with repositories.

As for the institutional repositories, we highlight the educational work called the Mexican Network of Institutional Repositories (REMERI by the acronym in Spanish), which emerged in 2012 with the purpose of integrating, spreading, preserving and giving visibility to the scientific, educational and documentary production of Mexico [20]. It is also the national network representing Mexico in the Federated Network of Institutional Repositories of Scientific Publications or LA REFERENCIA project where nine Latin American countries (Argentina, Brazil, Costa Rica, Chile, Colombia, Ecuador, El Salvador, Mexico and Peru) participate [21].

Also, the proposed law to create the national repository was approved during 2014 in Mexico. This proposed law is governed by the National Council on Science and Technology (CONACyT by the acronym in Spanish) and aims to make available the scientific and educational resources produced in Mexico with public funds [22].

In general, educational work based on semantic approach has examined proposals to exploit the Semantic Web approach from different perspectives. However, it is worth noting that there is little work developed in the establishment of educational guidelines to guide the process of creating OERs in the area of exercises, practices, thematic presentations, among others; since mainly OERs focus on the dissemination of theses, books and chapters of books, articles, etc., leaving aside teaching materials [23], and such materials represent the knowledge and experience that Professors have within the Institutions. On the other hand, an educational ontology that models education in the Mexican context has not been created. Besides, the data to describe OERs only include bibliographic metadata based mainly on the Dublin Core standard [24], without taking advantage of the content data and links to the resource itself.

3 Applied Methodology

The research presented is composed of two stages, these are: a) detection of needs by the study of use and preferences of Information and Communication Technologies (ICT) in the educational field, and b) building of the Prototype PMIIDAS.

3.1 Detection of Needs

At this stage, a study was carried out on the use of Information and Communication Technologies (ICT) to Professors and Learners of two higher education institutions. This study was applied to the curricula of Degree in Computing and the Degree in Electronics Engineering; as well as the Postgraduate in Information Science and Technology (master's and doctorate) attached to the Universidad Autónoma Metropolitana, Iztapalapa (UAM-I). Also, such study was applied on the Military Computing and Military Communications and Electronics Engineering of the Military School of Engineers (EMI by the acronym in Spanish).

From the analysis made to the study, user profiles, educational resources frequently used by Professors, educational resources of interest, and subjects requiring open educational resources could be determined. Using the information gathered from the study, we can propose the building of the prototype according to the needs expressed by the professors and learners of the two institutions.

3.2 Building of the Prototype MIIDAS

At this stage, different activities to be performed were established to develop the prototype, these are: a) Definition of the guidelines for generating educational resources; b) Building of the domain ontologies; c) Design of the prototype architecture, and d) Implementation of the prototype.

The following describes in detail each of the activities.

Definition of the guidelines for generating educational resources. In order to establish the guidelines for educational resources, the resources of greatest interest identified in the needs detection stage performed in both institutions were taken into account, which are: exercises, practices and thematic presentations.

Table 1. Common criteria for exercises, practices and thematic presentations (Source: own research).

Common criteria	Description
Educational Institution	Corresponds to the name of the educational institution to which the resource's author is attached.
Facilitator's Name	Name of the person who serves as the course instructor.
Name of Study Program	Name of the degree, engineering or postgraduate to which the resource impacts
Suggested Period of Use	Year, semester, quarter or quarter to which the resource can be applied.
Learning Unit	Name of the subject or subject for which the resource is designed.
Unit of Competence	Refers to one of the topics that are part of the Learning Unit.
Element of Competence	It is aimed at achieving the objective of the ongoing unit of competence.

Once the resources of greatest interest were identified, the educational approaches used by both educational institutions were explored, which was the competency-based approach. This approach is characterized by the identification of knowledge, expertise, skills, attitudes and values required to carry out a task [25].

The guidelines were worked together with specialists in pedagogy and didactics, by criteria that obey the competence approach. The common structure criteria among the three educational resources are presented in table 1.

It is important to point out that guidelines within institutions can help to standardize the educational resources created within the institution, as well as guiding new professors who are incorporated to generate their educational resources.

Building of the domain ontologies. Ontologies are a fundamental part of the Semantic Web Approach and therefore also for the Prototype PMIIDAS, since they provide a formal and standard representation of knowledge through a common vocabulary, favoring the exchange of such knowledge and its reuse among applications.

From the analysis carried out in the needs detection stage, 6 ontologies could be identified for the Prototype PMIIDAS. These are: user profile, domain (Computer, Electronic and Military), educational and digital educational resources. These ontologies have as main objective to establish a useful conceptual vocabulary for the project, as well as represent the concepts and relations inherent in the knowledge domain, thus constituting a knowledge model that allows us to make inferences about the data.

Methodology 101 [26] was adopted for building the ontologies. Such methodology was proposed by Stanford University and consists of 7 phases, which are: 1) Determine the domain and scope of the ontology; 2) Reuse existing ontologies; 3) List important terms for the ontology; 4) Define classes and their hierarchy; 5) Define the properties of the classes: slots, 6) Define the facets of the slots; and 7) Create instances or individuals. Table 2 shows the competence questions and the process to be performed for the user profile ontology.

Table 2. Competence questions and process to be performed for the user profile ontology (Source: own research).

Competence questions	Process to be performed
What professional activities do the professors and learners do? What skills do the professors and learners have? In what subjects are the professors specialized? What are the learners advised by professors? Who are the learners of a particular professor? What are the learners with a particular English reading domain and interest in Semantic Web research?	Adaptation of the user profile ontology proposed in [20] and the person ontology [11].

Table 3 and 4 show the competency questions and the process to be performed for the ontology of digital educational resources and the educational ontology, respectively.

The subjects requiring educational resources identified in the stage of needs detection were taken up again in order to determine the subjects and competency questions of the ontologies for computing and electronics. Table 5 shows the subject names identified by institution and user.

Table 3. Competence questions and process to be performed for the ontology of educational resources (Source: own research).

Competence questions	Process to be performed
What kind of educational resources are favorable for the active learning? What kind of educational resources are suggested for the subject of Database? What are the educational resources that have pdf extension? What are the English language resources for Electronics?	Adaptation of the ontology proposed in [11].

Table 4. Competence questions and process to be performed for educational ontology (Source: own research).

Competence questions	Process to be performed
What kind of educational approach do UAMI and EMI use? What are the teaching strategies established in the UAMI and the EMI? What educational resources can be used to support specific teaching strategies? What teaching strategies can help achieve the analysis? What are the assessment tools used by EMI?	The building of the ontology with a support of domain experts.

Table 5. Subjects requiring educational resources for UAM-I and EMI learners and professors (Source: own research).

User	UAM-I	EMI
	Subject	
Learners	Introduction to Programming Programming Fundamentals Data Structures and Algorithms	Electronics Programming Digital Systems Microcontrollers
Professors	Introduction to Programming Computer Networks Digital Communications Databases	Programming Electronics Microcontrollers Object-Oriented Software Development

Based on the identified subjects and the content analysis of each one, the competence questions are proposed for domain ontology for Computing and Electronics. Table 6 shows some of the competency questions and the activity to be performed.

In general, all the ontologies of the Prototype PMIIDAS have a total of 1,162 concepts distributed as follows: 22 concepts of user profile ontology; 30 concepts of the ontology for digital educational resources; 22 concepts of the Educational Ontology; and 804 concepts of the different domain ontologies distributed as follows: 662 concepts of computing; 142 concepts of Electronics; and 284 concepts of military ontology.

All the ontologies were developed with the ontology editor Protégé. On the other hand, it is important to point out that all ontologies have been validated by experts in each of the domains, as well as by the reasoner Pellet in order to verify the possible inconsistencies that could be found in them.

Table 6. Competence questions and process to be performed for ontologies for Computing, Electronics and Military (Source: own research).

Ontology	Competence question	Process to be performed
Computing	Is the star topology a network topology? Is the integer a primitive data type? Is analysis one of the stages in creating software?	Adaptation of the Ontology proposed in [4]
Electronics	What are the families of integrated circuits? What are the types of diodes? What are the parts of a transistor?	Adaptation of the Ontology proposed in [4]
Military	What is a council of war? What are the sub-values promoted by the Military Education System? What is a duty in the military field?	Building of the Ontology with the support of military personnel

Table 7. Use cases, description, user, and priority for the Prototype PMIIDAS (Source: own research).

User Case	Description	User	Priority
UC1-Login	Allows to log in to the prototype	Professor, learner and assessment committee	Half
UC2-Register user	Records the data of a new user	Professor, learner and assessment committee	Half
UC3-Generate OER	Creates OERs in digital form - exercise and practice type	Professor	High
UC4-Modify OER	Modify, delete and visualize the generated OERs	Professor	High
UC5-Load OER	Allows to load the OER into the IR, creating its semantic indexing	Professor	High
UC6-Recovering OER	Query and/or search for an OER	Professor, learner and assessment committee	High
UC7-Query guidelines	Shows the guidelines for the creation and use of OERs; as well as the prototype.	Professor, learner and assessment committee	Low

Prototype Design: Modeling is a central part of all the activities that lead to the production of good software. To carry out the requirements and architecture design activities for the Prototype PMIIDAS was based on the software architecture [27], where we used mainly the diagrams of 1) use cases and 2) architecture. The use of each is detailed below.

Use Case Diagram: These diagrams define the activities that the different users (professor, learner and assessment committee) will have in the prototype. The use cases, their description, the types of users, and the priority of each of them are showed in table 7.

Architecture Diagram: The type of architecture chosen for the creation of the Prototype PMIIDAS is a Web architecture due to the study carried out at EMI and UAM-I which considers the following reasons:

Users targeted by the prototype prefer to use web applications to query educational resources. In addition, they frequently use the computer for reviewing such resources.

Applications developed using a web architecture are lighter because they do not consume client resources, since the presentation, business and data layers are controlled by the server.

Fig. 1 shows the Architecture diagram of the Prototype PMIIDAS and different components thereof and the table 8 shows the description of the components of the Prototype PMIIDAS.

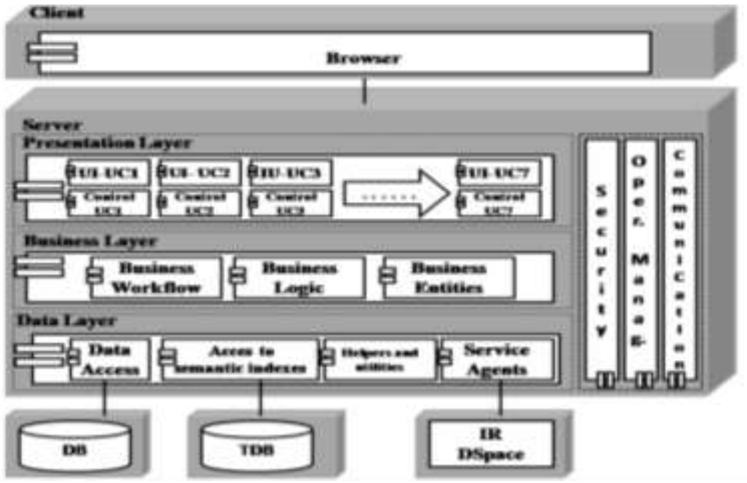


Fig.1. Architecture diagram of the Prototype PMIIDAS (notation: UML, Source: own research).

On the other hand, the Prototype PMIIDAS can also be seen as an integrator of different platforms, since it can superimpose a semantic layer where educational resources can be contained in multiple sites and be queried and retrieved by such prototype. Fig. 2 shows this concept, where a user makes a request for a resource, which is found on some platform, by a graphical interface. The Prototype PMIIDAS processes such request and together the semantic indexes know the location of this through the Unique Resource Identifier (URI). On the other hand, the Prototype PMIIDAS can consume the resources contained in the platforms provided there is a service provider [28].

Table 8. Description of the components that make up the architecture of the Prototype MIIDAS (Source: own research).

Component	Description
Browser	It is the application running on the client machine
UI-UC	It is the graphical interface in charge of the user interaction
UC Control	They are the components in charge of validations, business logics; as well as providing the data to the business layer
Workflow	It is the component in charge of managing the business processes
Business Logic	It is the component responsible for retrieving and processing data according to business logics
Business Entities	This component represents the business domain entities and their association with the business logic component
Access To Data	This component provides mechanisms to persist and retrieve data
Access To Semantic Indexes	This component is responsible for interacting with the Triple Store (TDB) to save and query the semantic indexes that characterize the resources
Tools and Utilities	This module contains functionality common with other modules such as: casting data types, e-mail sending, language coding, among others.
Service Agents	This module is responsible for the communication mechanisms to transfer data and resources to the institutional repository, such as DSpace
Security	Component responsible for authorization and authentication
Operation Management	This module handles exception management, instrumentation and validation
Communication	This module provides communication between the upper layers and physical levels

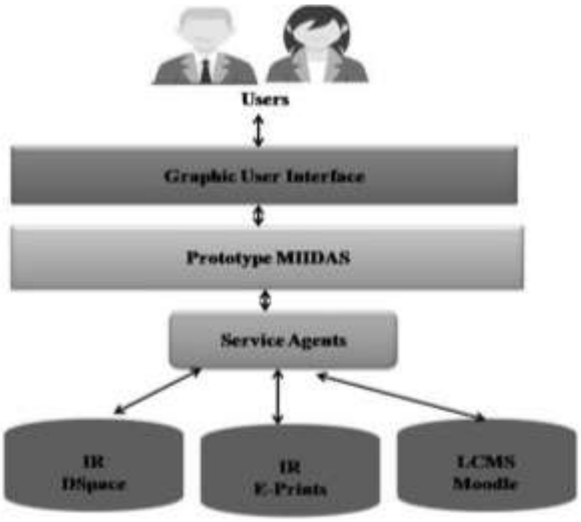


Fig. 2. Prototype PMIIDAS as integrator of different platforms (Source: own research).

Implementation. The building of the Prototype PMIIDAS is based on from analysis, design programming, data storage, educational resources and representation by semantic indexing free software languages and tools.

The prototype was developed in the Java programming language using the Apache Tomcat application server, as PostgreSQL Database Management System Version 9.4, Apache Jena Version 3.4 and DSpace Version 6.0. Fig. 3 shows the different technologies used to create the prototype.

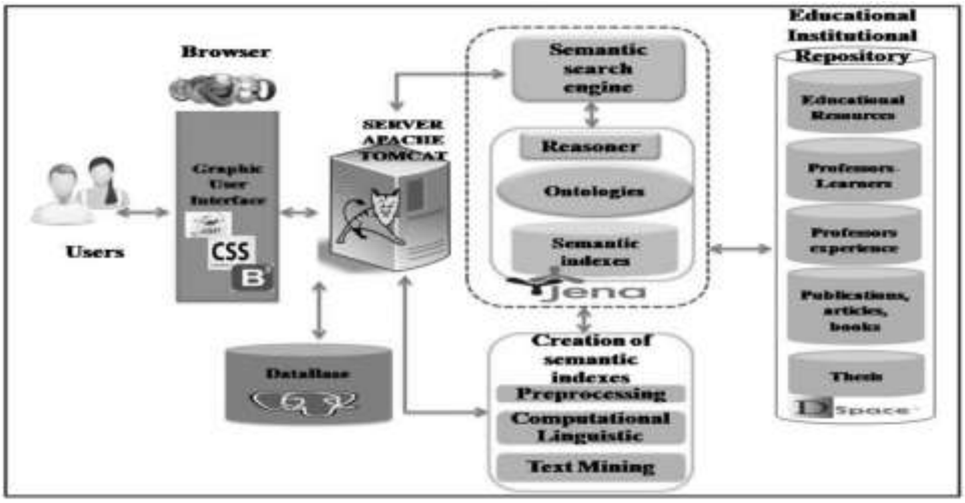


Fig. 3. Technologies used in the development of the Prototype PMIIDAS (Source: own research).

Where:

Interface. Allows communication between users and the system, is responsible for providing a visual environment to perform the different activities provided by the prototype according to the role of each user.

Apache Tomcat Server. Attends user requests from a graphical interface and answers according to the request.

Database. Conformed by user data and catalogues used in the creation of educational resources.

Semantic Search Engine. Recovers educational resources according to the search made by the use with the help of ontologies and semantic indexes.

Ontologies. Establishes the conceptual vocabulary of a field of knowledge (user profile, educational and domain resources: Computing, Military and Electronics) through concepts and relations thereof by constituting a model knowledge, which allows making inferences about the semantic indexing for educational resources.

Semantic Indexing (SMSI). Describes the educational resources made up of three parts: 1) metadata, 2) contents, and 3) links to others OERs.

Creation of SMSI. Creates the semantic indexing that describes each of the educational resources in Resource Description Framework (RDF), which are stored in the

Triple Store (TDB). The creation of semantic indexes is an automatic process, where the metadata is obtained from the form that fills the user to upload the file; to obtain the content and links of the same one is realized a preprocessing, to apply techniques of computational linguistics; as well as text mining.

Institutional Educational Repository: In charge of storing educational resources generated by professors in different formats and contents.

The prototype was developed under the usability quality attribute, which is to be friendly, intuitive and straightforward in navigation.

4 Experiments

The experiments carried out have been focused on testing the ontologies and the prototype.

4.1 Ontologies Testing

In general, the tests performed on the ontologies consisted of entering individuals and interrogating them using the competency questions initially made.

Table 9 shows the questions asked to the user profile ontology; as well as its translation in SPARQL query language and its result.

Table 9. Example of queries to the user profile ontology (Source: own research).

Natural Language Query	Query in SPARQL	Result

Who does Carolina Medina advise?	PREFIXopu:
	<http://pcyti.izt.uam.mx/pmiidas/ontoperfilusuario#>
	SELECT ?discente
	    WHERE {
	?docente opu:asesoraA ?discente.
	FILTER regex(?docente, "^Carolina Medina").}
		Cristal Galindo, Pablo Contreras

Who is an expert in Semantic Web?	SELECT ?docente
	    WHERE {
	?docente opu:esExpertoEn ?tema.
	FILTER regex(?tema, "web_semantica", "i"). }
		Carolina Medina, MaríaAuxilio Medina

Table 10 shows the questions asked to the ontology for digital educational resources. Table 11 shows the questions asked to the educational ontology.

4.2 Prototype Testing

The tests carried out on the Prototype PMIIDAS that have been made are: unitary and integration; As well as configuration. Each of them is detailed below.

Table 10. Example of queries to the ontology for digital educational resources (Source: own research).

Natural Language Query	Query in SPARQL	Result

What are the digital educational resources produced by UAMI?	PREFIX ored:
	http://pcyti.izt.uam.mx/pmiidas/ontored#
	SELECT ?recursosWHERE {
	?recursosored:esProducidoPor ?universidades. }
		Exercise 1
		Practice 5
		PresentationBD

What is the topic of Exercise 1?	SELECT ?temaWHERE {
	?recurso ored:tieneTema ?tema.
	FILTER regex (?recurso, "^Ejercicio1").}
		Programming

Table 11. Example of queries to the educational ontology (Source: own research).

Natural Language Query	Query in SPARQL	Result
To which educational institution belongs the Degree in Computing?	PREFIXoed: <http://pcyti.izt.uam.mx/pmiidas/ontoeduca#> SELECT ?ies WHERE { ?PEstudiooed:perteneceA ?ies. FILTERregex (?PEstudio, "^Licenciatura en Computacion").}	UAMI

To which model belongs the auditory learning style?	SELECT ?modelo
	WHERE {
	?eaoed:perteneceAModelo ?modelo.
	FILTER regex (?ea, "Auditivo" , "i"). }
		VAK

Unit and integration testing. Each module has been tested separately to verify the correct operation of the prototype. For this, test cases with repeatable, complete, reusable and independent characteristics were established.

On the other hand, the interaction among the different modules was verified. This test is known as an integration test.

Configuration testing. To discover specific bugs or compatibility issues in a particular environment, the prototype was tested in three Web browsers (Mozilla Firefox, Google Chrome, and Internet Explorer) with the intention of ensuring that the user experience is the same in all of them. Table 12 shows the bugs identified for each browser.

Table 12. Configuration test results (Source: own research).

Web Browser	Identified failures
Internet Explorer	The logo of the Prototype PMIIDAS in the header is traversed to the left The input help text does not appear
Google Chrome	Pop-up windows are not displayed
Mozilla	Calendars do not display The increment and decrement control for time does not work correctly

Regarding the usability of the Prototype PMIIDAS, an assessment tool (rubric) has been designed to evaluate educational software, which is based on [29] and [30]. This rubric is made up of 5 different sections, which are: 1) structure and display, 2) efficiency, 3) usability, 4) quality, and 5) pedagogical dimension. The first section aims to know the structure of the information in conjunction with the graphic elements. The second section seeks to understand the ease in which the users perform the tasks. The third section is intended to facilitate the interaction of the prototype with the user without having to consult the manual. The fourth section aims to know the accuracy of information retrieval. Finally, the fifth section seeks to know the specific characteristics to contribute to the teaching-learning process performed by the prototype.

The rating scale for each of the criteria is from 1 to 5, with 1 being the lowest rating and 5 being the highest rating.

5 Conclusions

This article has shown the methodology, design and implementation of the Prototype PMIIDAS, which allows the creation, edition, management and reuse of open educational resources that can be part of an institutional repository or any other platform. This is because it superimposes a semantic layer where each resource is characterized by a semantic index in RDF format and that point to the physical location of the resource.

This prototype was designed using the guidelines established by the software architecture and using the spiral model, because of this it was necessary to make different modifications at the level of the proposed design, the relational model, and the structure of the semantic index and technologies planned at the beginning of the process. The application offers the possibility to characterize different types of resources in other domains. Therefore, the prototype can be seen as a generic application to integrate resources in any domain. For which, it is only necessary to incorporate the ontologies according to the particular domain so that their applications and uses can be very diverse.

In addition, it is proposed the total implementation of the educational ontology which models education in the Mexican context, this to build applications that help in the preparation of primary and higher-level Professors to make the knowledge exam stipulated for the Professor assessment established by the National Institute for the Assessment of Education (INEE by the acronym in Spanish). On the other hand, it is suggested to perform tests with other institutional repositories other than DSpace; as well as the characterization of educational resources contained in Learning Content Management Platforms such as Moodle.

Acknowledgement. The authors would like to thank MS Laura Avilés Ortiz for her invaluable knowledge in Pedagogy, which was fundamental to build the educational ontology, Dr. Humberto Cervantes for his support and guidance in the architecture development of the Prototype PMIIDAS.

References

1. Budapest Open Access Initiative, http://www.budapestopenaccessinitiative.org/ (2017)
2. Bethesda Statement on Open Access Publishing, http://legacy.earlham.edu/~peters/fos/bethesda.htm (2017)
3. Berlin Declaration, https://openaccess.mpg.de/Berlin-Declaration (2017)
4. Álvarez Terrazas, J., Álvarez Terrazas, M., Gallegos Cereceres, V., Polanco Rodríguez, I.: La importancia de los repositorios institucionales para la educación y la investigación. Synthesis, 57(48), 43–48 (2011)
5. Van de Sompel, H., Lagoze, C.: The Santa Fe Convention of the Open Archives Initiative. D-lib Magazine, 6 (2) (2000)
6. Van de Sompel, H.: The Implementation of the Berlin Declaration on Open Access. D-lib Magazine, 11 (6) (2000)
7. The open archives initiative protocol for metadata harvesting, version 2.0. https://www.openarchives.org/OAI/openarchivesprotocol.html (2017)
8. OAI-PMH Data Provider 2.0 (Internals) - DSpace 6.x Documentation - DuraSpace Wik, Wiki.duraspace.org, https://wiki.duraspace.org/pages/viewpage.action?pageId=68064778 (2017)
9. OAI 2.0 Request Results, https://eprints.soton.ac.uk/cgi/oai2?verb=Identify (2017)
10. OPUS 4 Entwicklung. http://www.opus-repository.org/devdoc/ (2017)
11. Recursos educativos abiertos | Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura, Comunicación e información. http://www.unesco.org/new/es/communication-and-information/access-to-knowledge/open-educational-resources/ (2017)
12. Medina-Ramírez, R., Galindo Durán, C., García-Cué, J.: Hacia una gestión semántica, masiva, abierta y vinculada de conocimiento y recursos educativos. In Tópicos introductorios a la gestión del conocimiento, Ediciones de la noche, 103–125, México (2016)
13. Gertrudis-Casado, M., Gértrudix-Barrio, M., Álvarez-García, S.: Professional Information Skills and Open Data. Challenges for Citizen Empowerment and Social Change. Comunicar, 47, http://dx.doi.org/10.3916/C47-2016-04 (2016)
14. Aguirre, S., Quemada, J., SalvachuaJ.: Mediadores e Interoperabilidad en Elearning. V Conferencia Internacional Anual sobre educación, capacitación profesional y tecnologías de la información. Barcelona, España (2004)
15. Ibert, B., Silva, E., Soares, E.: A computational model for developing semantic web-based educational systems. Knowledge-Based Systems, 22 (4), 302–315. https://doi.org/10.1016/j.knosys.2009.02.012 (2009)
16. Arruti Gómez, A.: Analyzing and Producing Educational Resources for Didactalia.NET: A PilotProject Launched atthe University of Deusto (Spain Learners from Primary Education Degree). 8th International Technology, Education and Development Conference, Valencia, Spain (2014)
17. PROCOMÚN: Red de Recursos Educativos Abiertos, http://educalab.es/recursos/procomun (2017)
18. Abele, A., McCrae, J.: The Linking Open Data cloud diagram, Lod-cloud.net, http://lod-cloud.net/ (2017)
19. Medina Nieto, M., Sanchez, J., Cervantes, O., Benitez, A., De la Calleja, A.: LOD4AIR: A strategy to produce and consume linked open data from OAI-PMH repositories. International Conference on Electronics, Communications and Computers, CONIELECOMP, Puebla, Mexico (2017)
20. Red Mexicana de Repositorios Institucionales, http://www.remeri.org.mx/portal/index.html (2017)

21. LA Referencia Web Site, http://lareferencia.redclara.net/rfr/ (2017)
22. Estrategia de acceso abierto a la información científica, tecnológica y de Innovación Repositorios Institucionales. Términos de referencia para desarrollar los repositorios institucionales de acceso abierto a la información científica, tecnológica y de innovación, CONACYT, http://www.conacyt.gob.mx/index.php/el-conacyt/convocatorias-y-resultados-conacyt/convocatorias-direccion-adjunta-de-planeacion-y-evaluacion/convocatoria-2015-para-desarrollar-los-repositorios-institucionales-de-acceso-abierto-a-la-informacion-cientifica-tecnologica-y-de-innovacion/10724-terminos-de-referencia-repositorios-institucionales-2015/file (2017)
23. Documento del diagnóstico nacional de repositorios institucionales en la IES mexicanas mediante una muestra representativa, REMERI, http://www.remeri.org.mx/portal /img/documentos/Anexo3.pdf?iframe=true&width=100%&height=100% (2017)
24. DCMI Home: Dublin Core® Metadata Initiative (DCMI), http://dublincore.org/ (2017)
25. Frade Rubio, L.: Planeación por competencias. Inteligencia Educativa (2009)
26. Noy, N., McGuinness, D.: Ontology Development 101: A Guide to Creating Your First Ontology. Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880 (2001)
27. Cervantes, H., Kazman, R.: Designing Software Architectures. A Practical Approach. Addison-Wesley (2016)
28. Díaz, J., Schiavoni1, A., Amadeo, P., Osorio, A., Pietroboni, M., Pagano, M.: Integrando un Repositorio Digital con un Sistema de Gestión de Bibliotecas a través de OAI-PMH. XVII Workshop de Investigadores en Ciencias de la Computación, Salta (2015)
29. González Castañón, M.: Evaluación de software educativo: orientaciones para su uso pedagógico. http://www.tecnoedu.net/lecturas/materiales/lectura27.pdf (2017)
30. Rodriguez, A. P., Dominguez, E. L, Velazquez, H., M. Nieto, M. A.: Usability Assessment of Mobile Learning Objects by High School Learners. IEEE Latin America Transactions 14(2), 1044–1049 (2016)

ISSN 1870-4069

Measuring Influence on Twitter Using Text and User Relationships

Carlos Rodríguez, Gabriela Ramírez

Universidad Autónoma Metropolitana Unidad Cuajimalpa
Information Technologies Department, Cuajimalpa, Mexico

crodriguez@correo.cua.uam.mx, gramirez@correo.cua.uam.mx

Abstract. Graph theory concepts as centrality measure can be used to identify users, modelled as nodes of a graph, that have more influence or popularity in a social network. That can be used to classify users. Centrality is one of the most studied concepts in the analysis of Social Networks and there are a great variety of ways to measure it in order to identify the most relevant users in such networks. One of the main issues is how these measures can be calculated in a computationally tractable way and to allow users to be classified as closely as possible to reality. In the literature it can be found many interesting articles that study the application of the aforementioned measures in social networks with millions of users and an enormous amount of messages that flow in those networks. In the present article we are going to combine the information given by the mentioned graph theory measures with text analysis tools to improve the detection of influential users in the **Twitter** Social Network.

Keywords. Centrality measures, text analysis, user influence, social networks.

1 Introduction

Knowing the influence of users and being able to predict it can help to detect viral markets, improve searches, obtain recommendations from experts, more efficiently disseminate information or better manage social relationships with customers of a given company. In this paper we want to study how the influential on **Twitter** users can be detected. Given that the social networks can be modelled as graphs where the nodes represent the users and the edges represent the communication links among them, many graph theory tools become very useful for detecting who are the users that have the biggest audience, who are the users whose message are more cited, who issued the messages that are forwarded the most, etc. Many measures of influence have been presented in the literature ranging from those based on simple methods to those that appeal to complex mathematical models. Measures that record and differentiate between activity, such as popularity, are mentioned in such research works. The first article that we consulted about centrality measures in a network was [4]. In this article the authors studied the algorithmic aspect of calculating the betweenness centrality measure. Before the publication of [4] the algorithmic complexity of the best algorithm for calculating the betweenness centrality measure known at that time was $\Theta(n^2)$ in time and $\Theta(n^3)$ in space where n represented the number of actors in the network. Motivated by the fast growing of the social networks and the increased time for calculating the centrality measures on such networks, they were interested in calculating them efficiently. So their contribution [4] was to propose an algorithmic complexity improvement in time $O(nm)$ and in space $O(n + m)$ and for the case of weighted and unweighted their time complexity improvement was $O(mn + n^2 \log n)$ where m represented the number of links. With their algorithmic improvement they enlarged the range of networks for which the centrality analysis can be performed in an computationally tractable way. One of the articles about centrality measures applied to the subject of network efficiency that we consulted was [7]. The authors of [7] mention that the idea of structural centrality was applied with the end of characterise human communication in small groups of people and related this concept with the concept of influence in group processes. The authors of [7] introduced the information centrality measure, denoted as C^I in their paper. This measure is applicable in the case to groups and classes as well as in the case of individuals. The authors of [7] make the distinction between the *individual centrality* measure and the centrality based in the number of paths that pass through a node for reaching another node. Because of that they the notion of *information centrality* and related it with the notions of *degree centrality*, *closeness centrality* and *betweenness centrality* of the nodes, denoted as C^D, C^C and C^B respectively. In [2] the authors pointed out that Twitter is not so much a social network where a big number of participants are inactive accounts with low motivation to having dialogues. The authors of [2] say that the majority of the audience consumes and spreads the content published by small set influencer users, called alpha users, in a number of micro-networks. The authors of [2] say that the concept of the strength of weak ties is also applicable to Twitter, what means that the following users who are not part of a personal, strongly social network results in a greater amount of novel information. For this reason it is proposed in [2] a new and simple approach to measuring social networking potential (SNP) that combine content oriented ratio with a dialogue oriented ratio. The research purpose of [2] is to determine a grounded approach for measuring social networking potential of individual Twitter users. In the paper [3] studied the attributes and relative influence of 1.6M Twitter users and tracked 74 million diffusion events that took place on the Twitter follower graph during two month in 2009 and have found that the largest spreading of content tend to be generated by users who have been influential in the past and who have a large number of followers. The authors of [3] conclude that word-of-mouth diffusion can only be harnessed reliably by targeting large numbers of potential influencer. The authors of [3] obtain influencer information on Twitter by crawling the follower graph. In [5] the authors used a large amount of data collected from Twitter, we present an in-depth comparison of three measures of influence: indegree, retweets, and mentions and investigated the dynamics of user influence across topics and time. In [5] the authors observed that popular users who have high indegree are not necessarily influential in terms of spawning retweets or mentions. They also observed that most influential users can hold significant influence over a variety of topics. They concluded that topological measures such as indegree alone reveals very little about the influence of a user. Recently it was published in [9] a very complete surveys about centrality measures applied directly to the **Twitter** social network. The purpose of [9] is to collect and classify the different measures of influence of **Twitter** mentioning the ones based on the **PageRank** algorithm, those that use the content of the messages, others based on specific topics and others that try to make predictions. Additionally the mention some measures of activity and popularity, some mechanisms for correlating measures and some computational complexity aspects related to this context. The following are frequently used measures based on network topology: degree, closeness,betweenness, eigenvectors and eigenvalues of the adjacency matrix. The user and tweet relations are: user-to-user, user-to-tweet, tweet-to-user and tweet-to-tweet.

Metrics are simple mathematical expressions that provide basic information of a social network in numerical form. In bibliographical reference [9] the metrics involve: number of original tweets, number of *replies*, number of *mentions* and topological features of the network.

1.1 What does it Mean to be an Influential User?

This is a controversial topic because many criteria have been proposed as the ones that are innovative, prestigious, opinion leaders and authoritarian actors. Others associate them to being experts in a topic, opinion leaders, discussers or influencers about the opinions of others, inventors, disseminators, initiators of ideas and connectors. Thus they can be classified by the impact of their activity, diffusion capacity or by the content and authority of their messages. Other relevant users are em celebrities. They are classified by popularity in *broadcasters* and *in passives* (many *followers* and few *in followees*), contacts *acquaintances* # *in followers* ≈ # *in followees*) and *evangelists* (few *followers* and many *followees*) as is the case of *spammers* and *bots*. Some authors distinguish between being *in popular*, being *in influential*, *star* or *very read* by taking numerical metrics as the content of tweets. The author of [9] proposes to split the measures into three different types: activity measures, popularity measures (F1 better than F3) and influence measures (RT2 better than F2).

1.2 Activity Measures

The author of [9] consider that the active users are those who participate by sending: tweets, retweets, mentions and replies. For the calculation of the general activity it is proposed in [9] the following formula:

$$General\ Activity(i) = OT1 + RP1 + RT1 + FT1. \tag{1}$$

Table 1. Some important metrics on *Twitter*.

ID	Metric description
OT1	Number of tweets posted by the author
OT2	Number of shared URLs by his OTs
OT3	Number of hashtags included in their OT's
RP1	Number of RPs posted by the author
RP2	Number of tweets answered which conversation has been started by the author
RP3	Number of users that participated in RPs with the author
RT1	Number of RTs made by the author
RT2	Number of OT's posted by the author and retweeted by other users
RT3	Number of users that retweeted tweets of this author
FT1	Number of tweets marqued by others as favorites by the author
FT2	Number of tweets of the author marqued as favorites by other
FT3	Number of users that marcqued the tweets of the author as favorite
M1	Number of mentions by other user from the author
M2	Number of users mentioned by the author
M3	Number of mentions of the author by other users
M4	Number of users mentioning the author
F1	Number of followers
F2	Number of active followers in one subject
F3	Number of *followees*
F4	Number of active *followees* in one subject
F5	Number of followers sending tweets about a subject after the author
F6	Number of followers sending tweets about a subject before the author

1.3 Popularity Measures

A user is considered popular if it is recognized by many other authors on the network. A measure for this purpose is:

$$FollowerRank(i) = \frac{F1}{F1 + F3}, \tag{2}$$

There are variants of this measure such as em Tweeter Follower-Followee which is calculated as:

$$TTF(i) = \frac{F1}{F3}. \tag{3}$$

1.4 Influence Measures

According to the author of the article [9], an influential user is one whose actions in the network are able to affect the actions of other users in the network. Influential users tend to be active but few active users are influential. We can then think of some paradigms of social influence like: massive influence of a very persuasive little group or connected and accidental influence due to unpredictable factors.

1.5 Influential Users on a Topic

Some authors have been interested in studying influential users in a specific subject. Some traditional centrality measures used to measure influence on Twitter are α centrality and based on the *eigenvectors* using time t as an additional parameter and considering retweets. Another measure of influence proposed is that of *Information Diffusion* which estimates the possible influence of user tweets between *followers* (followed by *followers*). This measure is calculated as follows:

$$ID(i) = \log(F5 + 1) - \log(F6 + 1). \tag{4}$$

Many other user influence measures are mentioned in [9].

1.6 Aplications on the Web

Some application that runs in real time for the study of presidential elections and that has been applied to the detection of influential users in other social networks use : Data-mining, Text-mining, Graph theory based algorithms and Sentimient analysis. There are Web sites like *Klout, PeerIndex, InfluenceTracker, Twitter Grader, Favstar, BehaviorMatrix, Kred* or *Twitalyzer*, among others, to rank the most relevant Twitter users according to their activity, popularity or influence. Most applications measure global influences.

2 Text Analysis on Twitter

User influence on social media as Twitter, among other electronic social medias, has been object of study in sociology, communication, marketing, and political science. This notion is the basis for understanding how businesses operate. This same notion helps to understand how a small group of agents in a social network can change the opinion of the rest of the participants in a social network. If we are able to detect who are members of these small group of agents in a social network, we will be able to detect the opinion leaders, that is to say, those who can polarize the opinion on some topic in a discussion that takes place in a social network for the benefit of an advertising campaign. In this [8] the authors present an empirical analysis on opinion leaders identification problem in social networking medium as Twitter. The proposed approach for opinion leaders identification in [8] is based on the idea that the leadership/influential level of an author can be detected by considering its writing style, and its behavior within the Twitter community. According to this approach the authors of [8] propose several stylistics attributes (lexical richness, language complexity, etc), as well as different behavioral features (post's frequency, directed tweets, etc.), that are computed directly from users twitter accounts. When they have calculated all these features, they trained a classification model for identifying opinion leaders through machine learning algorithms and automatically identified influential users in a social network. The approach of [8] introduce the use text analysis techniques and behavioral features in order to detect the opinion leaders in a social network as Twitter. This work inspired us to propose a method for detection of opinion leaders that takes into account for this end, elements given by the text analysis in combination with the centrality measures mentioned in the introduction section of the present paper. In the next we will describe our proposed method.

3 Description of our Method

In this paper we propose that the identification of influential users on Twitter should not only be based on the analysis of metrics obtained of the user profile. From our point of view it must also be taken into account for this classification other features related to the style of reflected writing in their tweets and by the way the user interacts within the network with other users. The analysis of the metrics generated by a graph of relation of mentions of the user will give us greater elements to classify a user as influential or non-influential. For this en we will develop a web application, which allows users to be identified, on Twitter using textual attributes and attributes extracted from the graph of relationships between users. First we will implement a graph-based representation for a set related users on Twitter. After that we will Obtain and combine two types of attributes: attributes of the generated graph and shared text see [8] for more details. Then we will evaluate the identification of influential users with the use of a tagged collection. Finally we will build a web application that given a user name help us to determine if it is influential or not influential. For the purpose of this work we are going to consider a node (@userA) as a Twitter user and an edge as the relationship that is generated with another user (@ userB, @ userC, @ userD, etc) at the time @userA or @userB by means of a message or tweet, mention the other. No matter if whether @userA mentions @userB or @userB mentions @userA. In this sense, it is an unguided graph. Let's see the following figure where the graph is illustrated.

The literature identifies the following three types of influence for a user

1. Degree of influence, it refers the total number of followers, that is, the size of the audience.
2. Influence of retuits, that is the number of retuits that the user receives and indicates the amount of content a user generates which is transmitted through Twitter.
3. Mention influence, that measures the number of times a user is mentioned by others, indicating how many times this user initiates an interaction with other users.

In the article [8] it is proposed that the identification of influential users on Twitter should not only determined by these three parameters, but also by

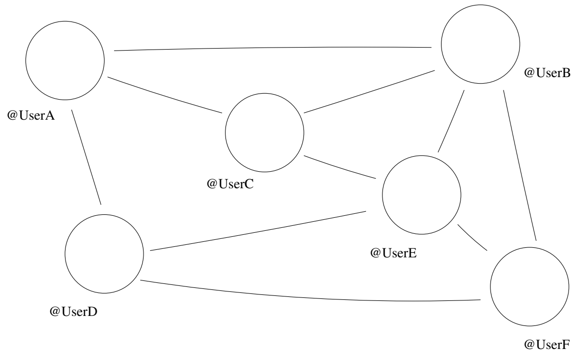


Fig. 1. Example of a non-directed graph on Twitter.

the style of writing and user behavior within the Twitter community as this is relevant to identify influential users. Thus, the influence of a user should not only be based on the analysis of the metrics of the user profile, in addition, the style of and how it interacts with other users makes it possible to identify more objectively those users capable of generating actions in others. We have called all the metrics we can get from a user's analysis of Twitter attributes. These attributes are numbered by the authors of [8] and are divided into two groups.

1. Style attibutes:
   (a) Words by tweet,
   (b) Size of words,
   (c) Length of the username,
   (d) Vocabulary Wealth,
   (e) Hapax,
   (f) Characters by tuit,
   (g) Size of user mentions,
   (h) Size of hashtags.
2. Behavioral Attributes:
   (a) User names in the description,
   (b) Number of hashtags in description,
   (c) URLs used in the description,
   (d) Self-mentions,
   (e) User age,
   (f) Number of tweets,
   (g) Number of followees,
   (h) Number of followers,
   (i) Shared multimedia content,
   (j) Number of favorites,
   (k) Followed by followers,
   (l) Tweets by followers,
   (m) Multimedia content per month,
   (n) URLs used in tweets,
   (o) Number of hashtags in tweets,
   (p) Direct messages,
   (q) Number of retweets,
   (r) Number of favorite tweets,
   (s) Frequency of tweets,
   (t) Standard deviation of the Frencuia by tweet.

To these attributes we add two more attributes *Closeness Centrality* (C_C) and *Betweenness Centrality* (C_B) retrieved from a graph that will be generated by the relationship between users denoted by the mentions that a given user makes other users. In such a way that the users of this graph will be the nodes and the edges will be the representation of the mentions that are between users.

4 Experimentation, Results and Evaluation

In this section we will talk about the obtained results from the training of the classification model and the results of the classification generated by the web application to different users. One of the main approaches of this work is that for the ranking of a Twitter user not only metrics from the profile of users are relevant, we can also use metrics extracted from a graph generated by the relationship between a user and those who are mentioned in their tweets, which will give us greater clarity of how it out the interaction within the network and therefore what is influence within it. The following tests are a result of the models with and without these metrics (C_C and C_B). For the training of the classification model in Weka and for the Web application, we use the Naive Bayes learning algorithm and the 10-fold cross validation test technique, which allows us to reduce the variance in the result. The 10-fold cross-validation consists of taking a test set and dividing it into 10 pieces, starting from one piece the other pieces are used nine to perform the tests, this is done 10 times, one for each one of the pieces and are saving the average of the 10 results. Finally, Weka runs the algorithm for the eleventh time with this data to generate the classification model. The results obtained by the Validation are shown below Crusade of 10 folds compared to the same test set of 250 users that we use for the training of the classification model but without the metrics of the graph.

Table 2. Comparison of 10-fold Cross Validation.

	Tot.Numb.	Correct Class	Incorrect Class	Rel. abs.	Prec.	Recall	F-measure
	of Instances	Instances	Instances	error			
No measures	250	137	113	108.1688	0.656	0.548	0.567
With measures	250	239	11	10.8838%	0.956	0.956	0.956

```
=== Confusion Matrix ===
a      b      a    b  <-- classified as
89     88   | 71    6  |      a = 0
25     48   |  5   68  |      b = 1
Non Graph     Graph
Metrics       Metrics
```

In the table 2 we show three measures evaluated by Weka:

1. **Precision** defined as the fraction of elements that really are classified as positive among all elements whose classification is defined as positive.
2. **Recall** the fraction of elements correctly classified as positive from all elements defined as positive.
3. **F-Measure** is simply the combination of the two previous measures:

$$\frac{2 \times Precision \times Recall}{(Precision + Recall)} .$$

As can be seen, the classification performed with the graph measures is more accurate and less susceptible to errors. On the other hand, the confusion matrix tells us how they were classified the 250 users, representing the value 0 the non-influent and 1 influent. The table 3 shows a training classification with the same 250 users. The results shown in the table 3 in the precision columns as absolute relative error indicate the reliability of the classification when the two additional metrics are used. As defined *F-Maesure* also gives us information on the reliability of the model. We can say, from these results that the classification is more precise if we include the two graph measures since as it was possible to observe in the last table, the instances or users classified was higher when graph measures were used than when these metrics were not considered. In a further test, we compared the classification of a user with the classification model. We compare the results obtained with and without the metrics of the graph.

Table 3. Training test.

	Tot.Numb. of Instances	Correct Class Instances	Incorrect Class Instances	Average precision	F-measure	Rel. error
No measures	250	117	73	0.501	0.587	100.011%
With measures	250	250	0	1	1	0.006%

Table 4. Classification of a user.

	Correctly classified of Instances	Precision	Recall	F-measure	Classsified as
No graph measures	1	1	1	1	0
With graph measures	1	1	1	1	0

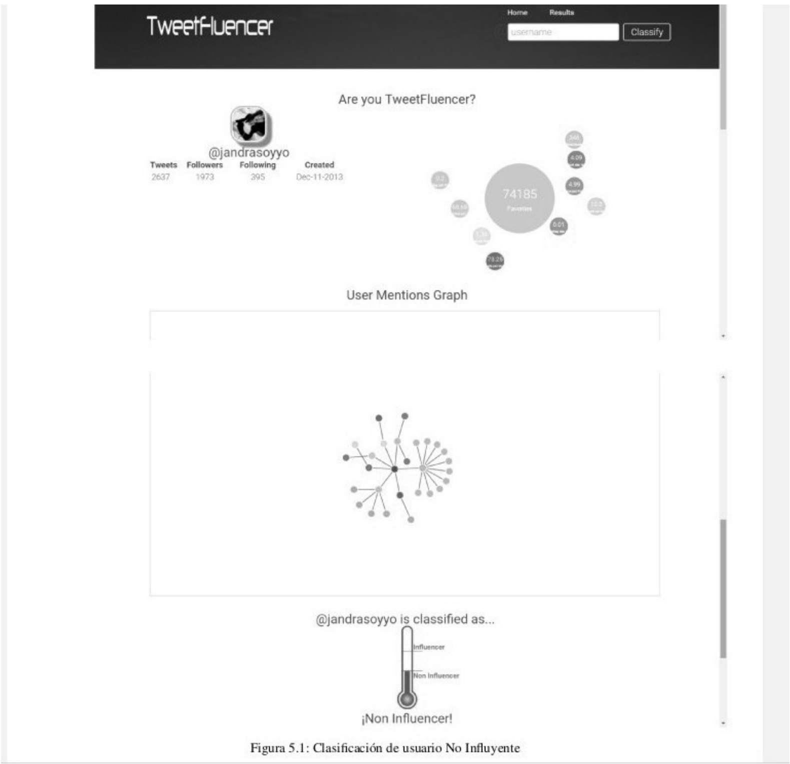
In this case, there was no difference between the two tests, the user was classified as Non-Influent denoted by a 0. It can be noticed that no difference is detected since only a new set has been classified. This leads us to conclude that although at the moment of classifying a single user there is no difference between using or not the metrics of the graph, at the moment of training the classifier model we can achieve greater precision if we include the graph measures.

5 Program Runs

The implementation of the Web Application consists of the development of modules that were worked sequentially to cover tasks required. The programs and libraries that were used are the following:

1. Php programming language.
2. Phyton programming language.
3. TwitterAPIExchange library for the connection with Twitter.
4. Weka library for the user classification model.
5. Python library for the treatise and analysis of the text of the tweets.
6. MatLab to identify the relationship between nodes.
7. HTML5 for the user interface.
8. D3js for the visualization of the relationship nodes between users.

With all of the above tests, we began classifications with the web application. In this section we show how is the visualization of the classification of users in the web application. Below is the result of the classification to two users, one was randomly chosen: @JandraSoyYo and a the other is well known political national figure: @EPN.



Are you TweetFluencer?

@jandrasoyyo

Tweets	Followers	Following	Created
2637	1973	395	Dec-11-2013

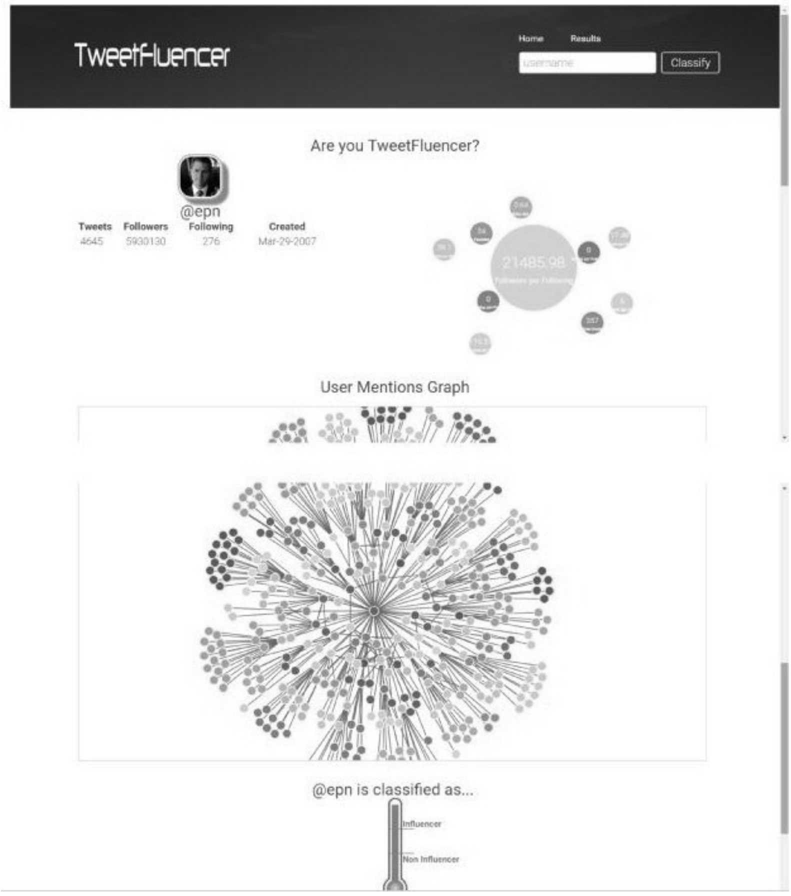
User Mentions Graph

@jandrasoyyo is classified as...

¡Non Influencer!

Figura 5.1: Clasificación de usuario No Influyente

Fig. 2. Twitter influence classification of user @jandrasoyyo as non-influencer.



Are you TweetFluencer?

@epn

Tweets	Followers	Following	Created
4645	5930130	276	Mar-29-2007

User Mentions Graph

@epn is classified as...

Fig. 3. Twitter influence classification of user @EPN as influencer.

6 Conclusions and Future Work

In the application of classification of influential users in Twitter, managed to classify a user of this social network based on the analysis of the text of your tweets, the relationship you have with other users and the main metrics obtained from your user profile. This gives us a more complete view of how a user is influenced by another within of the social network and the way in which complex relationships are woven among them. In order to carry out this analysis, we used a previously filtered and classified database which contained 2434 twitter users with 600 of their last tweets and information of the profile, from this one trained a classification model which served as base to compare the results obtained from this group with the results obtained from a new user, thus obtaining a classification based on the model.The classification obtained is not only based on the metrics obtained of a user's public profile, are also based on their writing style and the interaction they have with other users. At the time of training the classifier model we can achieve a lower degree of error if we include the metrics of the graph, therefore we can conclusion that our initial hypothesis where we assume that the analysis of the metrics generated by a relationship graph of user mentions will give us greater elements to classify a user as influential or non-influential is valid. Future work involves the following tasks:

1. Use different algorithms for classification and training model.
2. Add the results of the new classified users to the model of classification.
3. Reduce application process time, code debugging and error handling.
4. Improve graphical representation of the user mention graph, given that alredy the D3js library is robust and flexible, to generate a larger graph depth.

Acknowledgments. We want to acknowledge Carlos Geovany Pérez Velázquez for the programming implementation of the *Tweetfluencer* as part of his final project.

References

1. Aleahmad, A., Karisani, P., Rahgozar, M., Oroumchian, F.: Finding opinion leaders in online social networks Journal of Information Science 42, 1–16 (2015)
2. Anger, I., Kittl, C.:Measuring Influence on Twitter In: i-KNOW '11 Proceedings of the 11th International Conference on Knowledge Management and Knowledge Technologies,ACM N.Y. USA, Graz, Austria(2011)
3. Bakshy,E., Hofman,J.M., Mason,W.A., Watts,D.J.: Everyones an Influencer:Quantifying Influence on Twitter In: WSDM '11 Proceedings of the fourth ACM international conference on Web search and data mining, pp 65–74, ACM N.Y. USA, Hong Kong, China (2011)
4. Brandes, U.: A Faster Algorithm for Betweenness Centrality Journal of mathematical sociology Taylor and Francis 25, 163–177 (2001)
5. Cha, M., Haddadi, H., Benevenuto, F., Gummadi, K.P.: Measuring User Influence in Twitter: The Million Follower Fallacy, In: 4th International AAAI Conference on Weblogs and Social Media, pp 10-18.,Washington D.C. (2010)
6. Cossu, J.V., Labatut, V., Dugue, N.:A Review of Features for the Discrimination of Twitter Users:Application to the Prediction of Offline Influence, Cornell University Library,arXiv:1509.06585v1 [cs.CL] (2015)
7. Latora, V., Marchiori, M.: A measure of centrality based on the network efficiency, Cornell University Library, arXiv:cond-mat/0402050v1 [cond-mat.other] (2004)
8. Ramírez-de-la-Rosa,G.,Villatoro-Tello, E., Jiménez-Salazar, H., Sánchez-Sánchez, C.: Towards Automatic Detection of User Influence in Twitter by Means of Stylistic and Behavioral Features In: MICAI 2014 LNCS, vol. 8856, pp 245–256 Springer, Heildelberg (2014)
9. Riquelme, F.: Measuring user influence on Twitter: A survey, Cornell University Library, arXiv:1508.07951v1 [cs.SI] (2015)

ISSN 1870-4069

Evaluando un lexicón para la clasificación de polaridad a nivel de Tweet en español

Fabián Paniagua-Reyes, José A. Reyes-Ortiz, Belem Priego-Sánchez

Universidad Autónoma Metropolitana, Azcapotzalco,
México

{al2112002241,jaro,abps}@azc.uam.mx

Resumen. El análisis de sentimientos es una tarea que representa un reto y que cada día obtiene mayor relevancia en el procesamiento automático de textos. Sin embargo, existe una necesidad latente de contar con recursos de análisis de textos en español como la tarea de clasificación de polaridad. En este artículo se presenta una arquitectura para la evaluación de un lexicón denominado ML-SentiCon para la tarea de clasificación de polaridad en tweets en español, utilizando Máquinas de Soporte Vectorial, el modelo de bolsa de palabras y una ponderación de sus entradas mediante la métrica TF-IDF. La evaluación del lexicón se lleva a cabo en tres conjuntos de datos para diferentes cantidades de categorías. Los resultados de la evaluación muestran una perspectiva de comparación para los tres conjuntos.

Palabras clave: Lexicón, clasificación de polaridad, análisis de sentimientos, procesamiento de textos en español.

Evaluating a Lexicon for Polarity Classification at Tweet Level in Spanish

Abstract. The sentiment analysis is a task that represent a major challenge and, becomes increasingly relevant in text processing. However, there is a latent need to have resources for analyzing texts in Spanish such as polarity classification task. This paper presents an architecture to evaluate a lexicon called ML-SentiCon for polarity classification in Spanish tweets, using Support Vector Machine algorithm, the bag-of-words model and a weighting based on TF-IDF metric. The lexicon evaluation is carried out in three data sets for different number of categories. The evaluation results show a comparison perspective for such three sets.

Keywords. Lexicon, polarity classification, sentiment analysis, Spanish text processing.

pp. 179–189; rec. 2017-09-10; acc. 2017-10-20

1. Introducción

El análisis de sentimientos es una tarea que se encarga de crear modelos computacionales para procesar textos en lenguaje natural y determinar la carga emocional impresa en el texto. Este análisis abarca diversas tareas como: determinar la polaridad de un texto (positiva, negativa y neutra), identificar la emoción expresada en el texto (alegría, enojo, furia, tristeza, depresión) y establecer el grado de polaridad (muy positivo, positivo, negativo, muy negativo, neutro o nada). Estas actividades se abordan como una tarea de clasificación automática de texto.

El análisis de sentimientos es una tarea que se ha abordado, ampliamente, a partir de textos en inglés. Sin embargo, para el español las propuestas han sido escasas, esto ha ocasionado una carencia de recursos de análisis de opiniones para este idioma. En este aspecto, existen competencias internacionales para evaluar sistemas computacionales para el determinar la polaridad de un mensaje de red social.

Tal es el caso del Taller sobre Análisis Semántico en SEPLN (TASS) en sus diversas ediciones. En la edición del TASS 2017 [1] se presenta una tarea para el análisis de sentimientos a nivel de tweet y el análisis de sentimientos basados en aspectos. La tarea de análisis de sentimientos a nivel de tweet considera cuatro categorías: Positivo (P), Negativo (N), Neutro (NEU) y Ninguna (NONE). Mientras que en la edición del TASS 2016 [2], dicha tarea está catalogada en dos formas, por una parte, seis casos de etiquetas: Muy Positivo (P+), Positivo (P), Negativo (N), Muy Negativo (N+) Neutro (NEU) y Ninguna (NONE). Por otro lado, también existen los casos con cuatro etiquetas: Positivo (P), Negativo (N), Neutro (NEU) y Ninguna (NONE).

El uso de lexicones para la tarea de clasificación de la polaridad de un tweet es, cada día, más común. Los lexicones ofrecen una lista de palabras o lemas polarizadas con las cuales se puede determinar la polaridad de todo un mensaje o texto.

Po lo anteriormente descrito, en este artículo se presenta una arquitectura para evaluar el uso del lexicón denominado *ML-SentiCon* [3] en la tarea de clasificación de polaridad, utilizando los datos de la competencia TASS en sus ediciones 2016 y 2017 en la tarea de análisis de sentimientos a nivel de tweets en español. La tarea de clasificación se lleva a cabo con el algoritmo de Máquinas de Soporte Vectorial y como características, el pesado TF-IDF de los lemas en español del lexicón *ML-SentiCon.*

El resto del artículo se organiza como sigue: en la sección 2, se presenta el estado del arte de trabajos que han realizado el análisis de sentimientos en español con diversos enfoques. La sección 3, expone la arquitectura de evaluación del lexicón para la tarea de clasificación de polaridad. La sección 4, presenta la configuración experimental. Por su parte, la sección 5 expone los resultados de la experimentación realizada. La sección 6 presenta las conclusiones y el trabajo a futuro.

2. Estado del arte

El análisis de sentimientos para el español ha sido abordado desde diversos enfoques. Como en [4], [5] y [6], que han técnicas de aprendizaje profundo para representar los tweets y desempeñar el análisis de sentimientos a nivel de sentencias. El uso de vectores de características es ampliamente utilizado en la tarea de análisis de sentimientos. En [7] usan vectores de características de baja dimensión para representación del texto, ellos proponen un modelo simple fundamentado en la normalización de texto con identificación de marcadores de énfasis, el uso de modelos de lenguaje para representar las características locales y globales del texto, y características como emoticones y partículas de negación; en [8], del mismo modo, construyen vectores de palabras a partir de la información de opinión de recursos lingüísticos; en [9] representan los tweets por medio de vectores de palabras ponderados con TF-IDF y son clasificados utilizando algoritmos como máquinas de soporte vectorial (SVM) y regresión logística.

Las aproximaciones híbridas para el análisis de sentimientos también han sido empleadas, en [10] desarrollan una aproximación híbrida para el análisis de sentimiento global en Twitter, mediante el uso de clasificadores y aproximaciones sin supervisión, construidas mediante léxicos de polaridad y estructuras sintácticas.

El uso de lexicones para la tarea de clasificación de polaridad en textos de opiniones o mensajes de redes sociales, tanto en inglés como en español, se ha abordado por [11], quien desarrolló un lexicón con las palabras que co-existen en diversos corpora de opiniones y evalúa su desempeño con el algoritmo de Bayes. Por su parte, en [12] consideran como punto de partida el significado cambiante de una palabra que depende de las relaciones sociales que tiene cada autor, en dicho trabajo se utiliza una red neuronal para la tarea de clasificación. Finalmente, la adaptación de un recurso léxico denominado (ANEW) que originalmente no está destinado al idioma español, pero que [13] han traducido al español, es utilizada para identifican la necesidad de una opinión neutral al reorganizar la escala ANEW, si las palabras del conjunto de prueba no coinciden con el léxico generado, se consideran para la categoría de ninguno (NONE).

3. Descripción de la arquitectura

En esta sección se presenta una arquitectura (ver Figura 1) para evaluar el uso del lexicón denominado *ML-SentiCon* en la tarea de clasificación de polaridad a nivel de tweets en español.

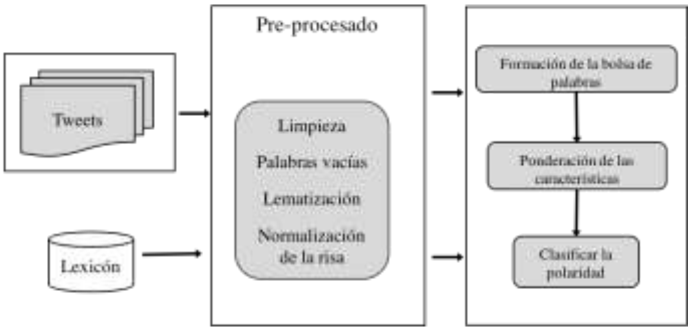


Fig. 1. Arquitectura para la evaluación del lexicón.

3.1. Lexicón

El lexicón evaluado en la arquitectura propuesta es denominado ML-SentiCon [3], el cual está constituido por 11302 entradas o lemas en español. Este recurso contiene lemas polarizados, con valores que van desde -1.0 "negativo" hasta +1.0 "positivo" y, adicionalmente, un valor de desviación estándar que refleja la ambigüedad resultante del cómputo de la polaridad a partir de los valores de los distintos significados posibles del lema. Además, se tiene la categoría gramatical para cada lema, a saber: verbo (v), sustantivo (s), adjetivo (adj), adverbio (adv), determinante (DT).

Este lexicón es generado de manera automática utilizando una versión mejorada del método usado para construir SentiWordNet 3.0 [14].

Las entradas (frases) del lexicón son extraídas, pre-procesada, representadas utilizando el modelo de bolsa de palabras y ponderadas. Ellas son utilizadas por el algoritmo de clasificación para determinar la polaridad del tweet.

3.2. Pre-procesado de tweets

Esta tarea de pre-procesado de los textos se aplica a los diversos experimentos realizados, así como, al lexicón utilizado. Por lo tanto, en esta sección se describen las tareas que involucra el pre-procesado de tweets.

La primera tarea es la limpieza de los textos. Para ello se realiza una segmentación por palabras (tokens) y la eliminación de caracteres especiales, como acentos (á, é, í, ó, ú) y signos de puntuación (. , ¡ ¿ ;). Después, las unidades léxicas son filtradas eliminando las ligas (url) a sitios web externos y las menciones de usuarios en Twitter (@). También, se lleva a cabo una normalización de las unidades léxicas resultantes a minúsculas y se eliminan las stopwords, palabras que no aportan significado y por lo tanto, no son funcionales para la identificación de polaridad. Sin embargo, se conservan las palabras de negación (no, ni) o afirmación (si), al ser consideradas como funcionales para la identificación de la polaridad manifestada por un tweet.

Las tareas de normalización de la risa y la lematización de las palabras son aplicadas a los experimentos con los lexicones externos y a los textos de los tweets con la finalidad de encontrar mayor correspondencia al momento de la ponderación de características.

El objetivo de la normalización de la risa es evitar la redundancia en la forma de expresarla. Para ello, se aplican las reglas o patrones mostrados en la Tabla 1, y se sustituyen las diversas formas de expresar risa por el término en común "jaja".

Tabla 1. Normalización de la risa.

Patrón	Frase	Risa normalizada
(ja)+	ja	jaja
(je)+	jeje	jaja
(jo)+	jojojo	jaja
(ji)+	jijijiji	jaja
lol	lol	jaja

La lematización consiste en obtener la raíz o lema de las palabras, es decir, eliminar los sufijos o flexiones de las palabras. Esto permite agrupar todas las palabras con la misma raíz en una sola representación en el lexicón propio y, mejora el mapeo de los términos de los lexicones externos con los textos de los tweets. Para esta tarea se utiliza la herramienta TreeTagger (Schmid, 1999), la cual tiene soporte para el español.

Tabla 2. Forma enraizada de palabras para el español.

Palabra	Lema
malas	malo
sentirán	sentir
fueron	ir
felices	feliz

En la Tabla 2, se ha mostrado un ejemplo de palabras en español tal como aparecen en los textos y con su lema (raíz) que es generada por la herramienta.

3.3. Extracción de características y ponderación

La extracción de características se realiza a partir de las entradas del lexicón ML-SentiCon pre-procesadas y su pesado es determinado con la métrica TF-IDF considerando los textos de los tweets. Tanto la extracción de características como la ponderación de las mismas se describen detalladamente como sigue.

Para la formación de características, se extraen las entradas del lexicón como unidades. Ellas representan las características para la etapa de clasificación de la polaridad de cada tweet en alguna categoría de acuerdo al conjunto de datos a evaluar. El modelo de bolsa de palabras (bag-of-words) es utilizado para la representación de las entradas de los lexicones.

La ponderación de las características está determinada por la métrica denominada Frecuencia del Término- Frecuencia Inversa en los Documentos (TF-IDF). Dicha métrica se basa en la frecuencia de aparición del término dentro de una colección de textos. Esta ponderación utiliza la frecuencia de aparición de los términos de la bolsa de palabras (lexicón) en un texto, la cual consiste en el número de veces que un término (*t*) del lexicón aparece en un *tweet* (*S*), ver Ecuación 1, y la frecuencia inversa que determina si el término es común en la colección de tweets (Ecuación 2). Esta información se utiliza, entonces, para calcular el valor final de *TF-IDF* (Ecuación 3):

$$TF(t_i,S_j) = f(t_i,S_j)\,, \tag{1}$$

$$IDF(t_i,S_j) = log\frac{|S|}{1+|s\in S\ :\ t_i\in s|}\,, \tag{2}$$

$$w_{ij} = TF(t_i, S_j) \times IDF(t_i, S_j). \tag{3}$$

En esta fase, se obtienen un conjunto de vectores numéricos que representan cada uno de los tweets del conjunto a evaluar, en este caso, las colecciones de tweets en español de la competencia TASS en sus ediciones 2016 y 2017.

3.4. Clasificación de la polaridad a nivel de tweet

La tarea clasificación de polaridad a nivel de tweets consiste en determinar la etiqueta (grado de polaridad) de un tweet basándose en su contenido. En esta arquitectura se lleva a cabo la evaluación del lexicón ML-SentiCon con tres conjuntos de datos (tweets en español).

Cada uno de estos conjuntos tienen un número diferente de categorías. Así pues, en su edición 2016 de la competencia del TASS, se proporcionan el mismo conjunto de tweets en dos versiones: una con seis casos de etiquetas: Muy Positivo (P+), Positivo (P), Negativo (N), Muy Negativo (N+) Neutro (NEU) y Ninguna (NONE) y la otra con cuatro casos de etiquetas: Positivo (P), Negativo (N), Neutro (NEU) y Ninguna (NONE). Además, la edición 2017 de TASS proporciona un nuevo conjunto de tweets con cuatro categorías: Positivo (P), Negativo (N), Neutro (NEU) y Ninguna (NONE).

La idea es utilizar un algoritmo de clasificación supervisada para determinar la etiqueta adecuada. Entonces, para esta tarea de clasificación supervisada se utiliza la representación de los tweets en los vectores ponderados con TF-IDF. Para dicha tarea, se utiliza el clasificador denominado Máquinas de Soporte Vectorial y presentado en [15], el cual construye un conjunto de hiperplanos en un espacio n-dimensional con los tweets de entrenamiento, estos hiperplanos son utilizados para predecir la clase de los tweets de prueba.

El clasificador basado en máquinas de soporte vectorial ha demostrado resultados alentadores en la clasificación de textos. La implementación del algoritmo de clasificación se ha llevado a cabo mediante la librería de aprendizaje automático denominada *scikit-learn* en Python [16].

4. Configuración experimental

El objetivo es evaluar el lexicón ML-SentiCon con tres conjuntos de tweets. Para ello, a partir del lexicón, se ha obtenido un conjunto de 3084 lemas o palabras que, después de ser ponderadas, forman una entrada en la matriz de la bolsa de palabras para cada conjunto de datos.

La evaluación del lexicón consiste en comparar los resultados para la tarea de clasificación de polaridad en tres conjuntos de tweets proporcionados por la competencia TASS, dos de la edición 2016 y uno de la edición 2017.

El conjunto de datos del TASS 2016 para seis etiquetas (P+, P, N, N+, NEU, NONE) consta de 7810 tweets de entrenamiento y 2577 tweets de prueba. Mientras que el conjunto de cuatro etiquetas corresponde al subconjunto de tweets con las cuatro etiquetas

(P, N, NEU, NONE), el cual consta de 6580 tweets de entrenamiento y 2172 tweets de prueba. Finalmente, el conjunto de datos del TASS 2017 consta de 7311 tweets de entrenamiento y 2400 tweets de prueba.

Todos los experimentos fueron llevados a cabo con los parámetros: complejidad o número de hiperplanos a construir: -C 1; parámetro gama (tipo de kernel a utilizar): -K PolyKernel; tamaño de la memoria cache a utilizar: -C 250007; parámetro de tolerancia: -L 0.001; y épsilon: -P 1.0E-12.

5. Resultados

Los resultados de la evaluación del lexicón para la clasificación de polaridad se centran en tres conjuntos de tweets: TASS 2016 con seis etiquetas (TASS 2016-6L), TASS 2016 con cuatro etiquetas (TASS 2016-4L) y TASS 2017 con cuatro etiquetas (TASS 2017-4L).

Los resultados de cada conjunto son presentados en términos de Precisión (P), Exhaustividad (R) y la medida F1. Estas métricas comparan los resultados del clasificador a evaluar con los valores externos de confianza (tweets etiquetados), utilizando los siguientes valores: a) Verdadero Positivo (VP) es el número de predicciones correctas del clasificador que corresponden al juicio externo de confianza (tweets etiquetados); Verdadero Negativo (VN) es el número de predicciones correctas del clasificador de polaridades que no corresponden al juicio externo de confianza; Falso Positivo (FP) corresponde al número predicciones incorrectas del clasificador que corresponden al juicio externo de confianza; y, finalmente Falso Negativo (FN) es el número de predicciones incorrectas del clasificador que no corresponden al juicio externo de confianza.

Bajo estos criterios, se emplea la Precisión (P) para evaluar los algoritmos en términos de los valores de predicciones positivas, la cual se define, en la Ecuación 4, como:

$$P=\frac{VP}{VP+FP}\,. \tag{4}$$

También, se utiliza el Exhaustividad (R) para expresar la tasa de correspondencias correctas con las opiniones de textos preclasificados de manera externa con una alta confianza (Ecuación 5):

$$R=\frac{VP}{VP+FN}\,. \tag{5}$$

Finalmente, la medida F (F1) que representa la media armónica entre Precisión y Cobertura, la cual tiene como fundamento el obtener un valor único ponderado entre ellas (Ecuación 6):

$$F1=2*\frac{P*R}{P+R}\,. \tag{6}$$

Los resultados de evaluar el lexicón ML-SentiCon para el conjunto TASS 2016 con seis etiquetas (TASS 2016-6L), en la tarea de clasificación de polaridad, se muestra en la Tabla 3.

Tabla 3. Resultados para el conjunto TASS 2016-6L.

Etiqueta	P	R	F1
P+	0.508	0.506	0.507
P	0.488	0.489	0.488
N	0.433	0.430	0.431
N+	0.413	0.411	0.412
NEU	0.375	0.373	0.374
NONE	0.103	0.104	0.103

Los resultados de evaluar el lexicón para el conjunto TASS 2016 con cuatro etiquetas (TASS 2016-4L), se muestra en la Tabla 4.

Tabla 4. Resultados para el conjunto TASS 2016-4L.

Etiqueta	P	R	F1
P	0.540	0.540	0.540
N	0.529	0.528	0.529
NEU	0.497	0.495	0.496
NONE	0.165	0.163	0.164

Los resultados de evaluar el lexicón para el conjunto TASS 2017 con cuatro etiquetas (TASS 2017-4L), se muestra en la Tabla 5.

Tabla 5. Resultados para el conjunto TASS 2017-4L.

Etiqueta	P	R	F1
P	0.553	0.551	0.552
N	0.539	0.536	0.537
NEU	0.509	0.509	0.509
NONE	0.168	0.166	0.167

La categoría de tweets con la etiqueta NONE tuvo un pobre desempeño en los tres conjuntos de datos, considerando el lexicón ML-SentiCon. Esto se debe, en gran medida, a que dicho lexicón carece de palabras para esta categoría, es decir, sin carga de polaridad o sin valor. Todas las entradas en el lexicón tienen un valor entre -1 y +1.

Finalmente, la Tabla 6 muestra la comparativa que permite evaluar el lexicón en los tres subconjuntos: TASS 2016 con seis etiquetas (TASS 2016-6L), TASS 2016 con cuatro etiquetas (TASS 2016-4L) y TASS 2017 con cuatro etiquetas (TASS 2017-4L). Estos resultados son mostrados utilizando las versiones macro (promedios) de Precisión (Macro-P), Exhaustividad (Macro-R) y medida F1 (Macro-F1) con la finalidad de proporcionar una comparación comprensiva de los resultados.

Tabla 6. Resultados del lexicón en tres conjuntos.

Conjunto de datos	Macro-P	Macro-R	Macro-F1
TASS 2016-6L	0.387	0.375	0.386
TASS 2016-4L	0.433	0.431	0.432
TASS 2017-4L	0.442	0.441	0.441

Los resultados mostrados en la Tabla 6 hacen constar que el lexicón ML-SentiCon tiene un mejor desempeño, en cuanto a la tarea de clasificación de polaridad, en el conjunto de la competencia TASS en su edición 2017 considerando cuatro etiquetas para los tweets: Positivo (P), Negativo (N), Neutro (NEU) y Ninguna (NONE).

6. Conclusiones

Este artículo ha presentado una arquitectura para evaluar el uso del lexicón denominado ML-SentiCon en la tarea de clasificación de polaridad en mensajes de tweets en español. La arquitectura presentada considera como componente al lexicón y lo evalúa en tres conjuntos de tweets: TASS 2016 con seis etiquetas, TASS 2016 con cuatro etiquetas y TASS 2017 con cuatro etiquetas. La clasificación de la polaridad se lleva a cabo con el algoritmo Máquinas de Soporte Vectorial utilizando un modelo de bolsa de palabras y una ponderación TF-IDF.

La evaluación del lexicón ha hecho evidente que, en los tres conjuntos de datos, la categoría de NONE presente los peores resultados en cuanto a medida F1. Esto se debe, en gran medida, a que el lexicón no contiene entradas para dicha categoría. Es decir, todas las palabras tienen un valor de polaridad asignado.

Con las pruebas de evaluación del lexicón, también, ha resultado que el lexicón ML-SentiCon tiene un mejor desempeño, en cuanto a la tarea de clasificación de polaridad, en el conjunto de la competencia TASS en su edición 2017 que con el resto de los conjuntos de tweets.

Este trabajo ha contribuido en contar con un recurso de análisis comparativo de recursos lingüísticos para el español en la tarea de análisis de sentimientos.

Una línea de investigación como trabajo a futuro, se puede centrar en la categoría de NONE y proponer un enfoque para considerar entradas en los lexicones sin polaridad. Como trabajo futuro, también, se puede experimentar con más lexicones con entradas polarizadas o categorizadas (Positivas Negativas, Neutras) para el español y extender la evaluación hacia otros conjuntos de tweets. Finalmente, considerar la clasificación de tweets por tema o tópico considerando el lexicón ML-SentiCon, puede resultar de gran utilidad.

Agradecimientos. Este trabajo fue apoyado, parcialmente, por la Red Temática en Tecnologías del Lenguaje de CONACyT. Los autores agradecen a la Universidad Autónoma Metropolitana unidad Azcapotzalco por el apoyo recibido para la realización de este trabajo.

Referencias

1. Martínez-Cámara, E., Díaz-Galiano, M. C., García-Cumbreras, M. Á., García-Vega, M., Villena-Román, J: Overview of TASS 2017. In: Proceedings of TASS 2017: Workshop on Semantic Analysis at SEPLN (TASS 2017), CEUR Workshop Proceedings, España, 13–21 (2017)
2. García-Cumbreras, M. Á., Villena-Román, J., Cámara, E. M., Díaz-Galiano, M. C., Martín-Valdivia, M. T., López, L. A. U.: Overview of TASS 2016. In: Proceedings of TASS 2016: Workshop on Sentiment Analysis at SEPLN colocated with 31st SEPLN Conference. CEUR Workshop Proceedings, España, 13–21 (2016)
3. Cruz, F. L., Troyano, J. A., Pontes, B., Ortega, F. J.: ML-SentiCon: Un lexicón multilingüe de polaridades semánticas a nivel de lemas. Procesamiento del Lenguaje Natural, 53, 113–120 (2014)
4. Vilares, D., Doval, Y., Alonso, M. A., Gómez-Rodríguez, C.: LyS at TASS 2015: Deep Learning Experiments for Sentiment Analysis on Spanish Tweets. In: Proceedings of TASS 2015: Workshop on Sentiment Analysis at SEPLN co-located with 31st SEPLN Conference (SEPLN 2015), CEUR Workshop Proceedings, España, 47–52 (2015)
5. Díaz-Galiano, M. C., Montejo-Ráez, A.: Participación de SINAI DW2Vec en TASS 2015. In: Proceedings of TASS 2015: Workshop on Sentiment Analysis at SEPLN co-located with 31st SEPLN Conference (SEPLN 2015), CEUR Workshop Proceedings, España, 59–64 (2015)
6. Montejo-Ráez, A., Díaz-Galiano, M. C.: Participación de SINAI en TASS 2016. In: Proceedings of TASS 2016: Workshop on Sentiment Analysis at SEPLN colocated with 31st SEPLN Conference, CEUR Workshop Proceedings, España, 41–45 (2016)
7. Murillo, E. C., Raventós, G. M.: Evaluación de Modelos de Representación del Texto con Vectores de Dimensión Reducida para Análisis de Sentimiento. In: Proceedings of TASS 2016: Workshop on Sentiment Analysis at SEPLN co-located with 31st SEPLN Conference, CEUR Workshop Proceedings, España, 23–28 (2016)
8. Martínez-Cámara, E., García Cumbreras, M. Á., Martín-Valdivia, M. T., Ureña-López, L. A.: SINAI-EMMA: Vectores de Palabras para el Análisis de Opiniones en Twitter. In: Proceedings of TASS 2015: Workshop on Sentiment Analysis at SEPLN co-located with 31st SEPLN Conference (SEPLN 2015), CEUR Workshop Proceedings, España, 41–46 (2015)
9. Quirós, A., Segura-Bedmar, I., Martínez, P.: LABDA at the 2016 TASS Challenge Task: Using Word Embeddings for the Sentiment Analysis Task. In: Proceedings of TASS 2016: Workshop on Sentiment Analysis at SEPLN co-located with 31st SEPLN Conference, CEUR Workshop Proceedings, Españ, 29–33a (2016)
10. Alvarez-López, T., Juncal-Martínez, J., Gavilanes, M. F., Costa-Montenegro, E., González-Castano, F. J., Cerezo-Costas, H., Celix-Salgado, D.: GTI-Gradiant at TASS 2015: A Hybrid Approach for Sentiment Analysis in Twitter. In: Proceedings of TASS 2015: Workshop on Sentiment Analysis at SEPLN co-located with 31st SEPLN Conference (SEPLN 2015), CEUR Workshop Proceedings, España, 35–40 (2015)
11. Eisenstein, J.: Unsupervised Learning for Lexicon-Based Classification. In: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17), AAAI Publications, USA, 3188–3194 (2017)
12. Yang, Y., Eisenstein, J.: Overcoming Language Variation in Sentiment Analysis with Social Attention. Transactions of the Association for Computational Linguistics 5, 259–307 (2017)

13. Henriquez, M., Guzman, J. A., Salcedo, D.: Minería de Opiniones basado en la adaptación al español de ANEW sobre opiniones acerca de hoteles. Procesamiento del Lenguaje Natural, 56, 25–32 (2016)
14. Baccianella, S., Esuli, A., Sebastiani, F.: Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In: Proceedings of the Seventh conference on International Language Resources and Evaluation (ELRA), Malta, 2200–2204 (2010)
15. Chang, Ch., Lin, Ch.: LIBSVM - A Library for Support Vector Machines. ACM Transactions on Intelligent Systems and Technology 2(3), 27–28 (2001)
16. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M. Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 12, 2825–2830 (2011)

ISSN 1870-4069

Diseño de escenarios de aprendizaje con interfaces naturales y realidad aumentada para apoyar la inclusión de estudiantes con discapacidad auditiva en la educación media superior

Carmen Cerón, Etelvina Archundia, Alfonso Garcés, Beatriz Beltrán, Jair Migliolo

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

{mceron,etelvina,agarces,bbeltran}@cs.buap.mx, jair@gmail.com

Resumen. El propósito de este artículo es presentar el diseño de un sistema interactivo con escenarios de aprendizaje para apoyar a estudiantes con discapacidad auditiva de educación media superior en el área de biología. La metodología utilizada fue de prototipos y el diseño centrado en el usuario, lo cual permitió el uso del dispositivo natural del kinect® y la RA para dispositivos móviles como tabletas y Smartphone con Android. Finalmente se presentan los resultados obtenidos de una prueba piloto a un grupo focal de estudiantes con discapacidad auditiva, la experiencia de usuario y usabilidad del sistema.

Palabras claves: interfaces naturales de usuario, discapacidad auditiva, ambientes virtuales de aprendizajes.

Design of learning scenarios with natural interfaces and augmented reality to support the inclusion of students with hearing disabilities in upper secondary education

Abstract. The purpose of this paper is to present the design of an interactive system with learning scenes with augmented reality for helping to support hearing impaired students of high school in biology area. The methodology used was Prototypes and the Centered Design of the User, which allowed the use of the natural device of Kinect® and AR for mobile devices like tablets and smartphones with Android. Finally, it is presented the achieved results from a pilot test to a focal group of hearing impaired students, the user experience and the usability of the system.

Keywords: natural user interfaces, auditory disability, learning virtual environment.

pp. 191–201; rec. 2017-08-03; acc. 2017-10-05

1. Introducción

La tecnología digital se ha convertido una herramienta importante para las personas con discapacidad, ya que les permiten acceder a diferentes entornos: recreación, educación, trabajo, comunicación e información. La Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura (UNESCO) afirma que la “inclusión se orienta a transformar los sistemas educativos para responder a la diversidad del alumnado y a una educación con igualdad de oportunidades: acceso, permanencia, participación y los logros de todos los estudiantes, con especial énfasis en aquellos que, por diferentes razones, están excluidos o en riesgo de ser marginados” [1]. Tal es el caso de personas con discapacidad, indígenas y niños. Es por eso que existe la necesidad de incidir e integrar a estudiantes con discapacidad en la educación media superior, logrando ser inclusivos mediante entornos de aprendizajes enriquecidos para todos. Una de las necesidades que se han identificado en la educación media superior para lograr la inclusión de esta población es la de propiciar espacios, aulas, desarrollar materiales didácticos, ayudas de software y recursos que puedan utilizarse en la modalidad presencial de forma accesible y permitan enriquecer el aprendizaje de los estudiantes.

El propósito de esta investigación es presentar el diseño de un sistema interactivo SIM’s Anatomy con escenarios de aprendizaje accesibles por medio de Interfaces Naturales de Usuario mediante el uso del dispositivo del Kinect®, para facilitar la estimulación física y cognitiva, mediante uso de señas, movimientos y reconocimiento postural del cuerpo humano. Así también, el uso de escenarios en dispositivos móviles Smartphone o Tabletas con sistema Android integrando realidad aumentada para la representación de las actividades de aprendizaje e interacción, adaptándose a las necesidades y limitaciones de los usuarios para lograr el acceso al sistema, siendo atractivo y amigable acerca de temas relacionados con Tópicos de Biología: Cuerpo Humano, llamado esta versión de móviles como sistema AR’Anatomy. Este sistema permitirá apoyar el proceso aprendizaje y desarrollo de competencias disciplinares en los estudiantes con discapacidad auditiva mediante actividades de aprendizaje interactivas con RA.

El artículo está organizado como se indica a continuación: en la sección 2 se presenta la fundamentación y la revisión teórica del trabajo de las Interfaces Naturales de Usuario y experiencias de aprendizaje de realidad aumentada. La sección 3 se describe el análisis y el diseño del sistema, la arquitectura e implementación del sistema con las tecnologías de Kinect® y de los dispositivos móviles. En la sección 4 se muestran los resultados de la prueba piloto del grupo focal con el prototipo del sistema. Finalmente, se presenta las conclusiones y el trabajo a futuro de esta investigación.

2. Estado del arte

En esta sección se revisan aspectos de discapacidad auditiva, las interfaces naturales de usuario y la realidad aumentada, así como diferentes experiencias aplicadas al campo de la educación, los cuales aportan fundamentos para este trabajo de investigación.

2.1. Discapacidad auditiva e implicaciones educativas

La discapacidad auditiva es la dificultad o imposibilidad que presenta la persona al utilizar el sentido del oído. En términos de la capacidad auditiva, se detectan dos conceptos relacionados, la hipoacusia y sordera. La primera se refiere a la pérdida auditiva de leve a moderada; no obstante, resulta funcional para la vida diaria. Por otra parte, la segunda es conocida como sordera se refiere a la pérdida auditiva de severa a profunda, donde la audición no es funcional para la vida diaria aun con auxiliares auditivos; la adquisición del lenguaje oral no se da de manera natural [2]. *Los alumnos con este tipo de perdida utilizan principalmente el canal visual, para recibir la información, aprender y comunicarse*. Por lo que la Lengua de Señas es la lengua natural de las personas con esta condición y que requieren estrategias específicas por los docentes para ser atendidos en los distintos niveles educativos. Por consiguiente, la tecnología y las interfaces naturales con distintos dispositivos ofrecen una gama de posibilidades para apoyar y lograr una mayor interacción con estudiantes con tal discapacidad. Los grados de pérdida auditiva según Villalba [3] se muestran en la Tabla 1. Es importante reconocer las implicaciones que conlleva el reconocimiento de sonidos, la lengua de señas y comunicación, lo que hace necesario ciertas recomendaciones para el docente en cada grado de pérdida auditiva y que debe propiciar estrategias para el alumno.

Tabla 1. Grados de pérdida auditiva y recomendaciones educativas.

Pérdida Auditiva	Implicaciones	Recomendaciones
Leve (20 a 40 dB)	No captan fragmentos con "s o t" o una voz débil. Se distraen con facilidad y requieren se les repita lo que escuchan.	Apoyarse con estrategias visuales con un orden lógico y secuenciado. Repetir puntos principales.
Moderada (41 a 70 dB)	Solo escucha con voz fuerte. Se apoyan en la lectura de labio facial. Usan lenguaje tardío y fonemas. Problemas en su estructuración del lenguaje y bajo vocabulario.	Aplicar técnicas de colaboración en grupos pequeños. Utilizar ayudas técnicas digitales y permitir el manejo de comunicadores visuales y parlantes.
Severa (71 a 90 dB)	Presentan problemas del lenguaje y se aíslan de las personas. Requiere el aprendizaje de Lenguaje de Señas o interpretación de labio facial	Manejo de Lenguaje de Señas Ayudas técnicas digitales Materiales audiovisuales Estar siempre de frente al alumno Trabajar en pareja.
Profunda (más de 90 dB) y Sordera Total	Falta de un código de comunicación. Confía más en lo visual y debe estimular la percepción del tacto y visual. No escuchan. Voz distorsionada o sonidos guturales emiten	Uso de Lenguaje de Señas Estrategias del Bilingüismo Evitar discriminaciones Utilizar ayudas técnicas digitales y dispositivos con mayor interacción visual.

2.2. Interfaces naturales y aplicaciones

Las interfaces naturales de usuario (NUI) son aquellas en las que se interactúa con un sistema, aplicación, etc., sin utilizar sistemas de mando o dispositivos de entrada de las GUI (Interfaces de Usuario Graficas), como sería el ratón, teclado alfanumérico, joystick, etc., y en su lugar, se hace uso de movimientos gestuales tales como las manos o el cuerpo que fungen como el mismo mando de control, algunos ejemplos de estos dispositivos naturales son: kinect sensor, leap motion controller, myo gesture control armband y Google Glass. En esta investigación nos enfocamos al dispositivo Kinect®.

2.2.1. Dispositivo Kinect

Microsoft Kinect Sensor, es un dispositivo que integra una gran cantidad de sensores para detectar movimiento corporal del usuario, al mismo tiempo captura el sonido permitiendo la interacción utilizando instrucciones de voz [11]. Es un sensor de bajo costo que permite reconocer los movimientos del cuerpo y del rostro del usuario eliminando así el uso de controles físicos. Se compone de 2 sensores de profundidad 3D, cámara RGB, arreglo de micrófonos, y un motor automático que ajusta la base. Reconoce hasta 20 puntos del cuerpo como son la cabeza, hombros, manos, etc. y se programa SDK (Software Development Kit) en C/C++, C#, VB y librerías como OpenCV. Por lo que, como medio de reconocimiento de movimiento, de las manos y algunas señas específicas, se logra ejercitar procesos cognitivos y permitir que la persona con discapacidad auditiva utilice las interfaces naturales como un código de comunicación con los escenarios de aprendizaje [4].

2.2.2. Experiencias con interfaces naturales

En el proyecto Mudra [5], su objetivo fue utilizar como dispositivos unos guantes con marcadores de colores y un sensor de profundidad para simular las operaciones del dispositivo del ratón en el uso del sistema operativo. Por otra parte, en el área de la salud, un estudio propuesto para analizar y realizar un seguimiento sobre las posturas y movimientos que realizan los usuarios al colocarse en diversas posiciones y acciones motrices usando Kinect® para mejorar su salud física y mental [6]. Así también, en otra investigación, el uso de una interfaz natural como el Kinect® con carácter rehabilitador se ha utilizado en varios usuarios con problemas motrices en el que demuestran que el uso de interfaces naturales puede ayudar a la rehabilitación de personas con problemas de carácter motriz y aumenta la mejora notablemente frente a otros métodos de rehabilitación convencionales [7].

Las Nuevas Tecnologías (NT) cumplen dos funciones básicas especialmente vinculadas con el aprendizaje: *la mediación cognitiva y la aceleración de la percepción estímulos sensoriales.* Un escenario de aprendizaje apoyado en la tecnología debe ser enriquecido para incrementar la eficiencia e interactividad con los contenidos de aprendizaje, ya que la tecnología propicia la interacción de alto nivel y promueve el desarrollo de habilidades del pensamiento crítico, creativo y manejo espacial [8].

2.3. Realidad Aumentada

La Realidad aumentada (RA) concebida como “la generación de imágenes nuevas a partir de la combinación de información digital en tiempo real y el campo de visión de una persona” [9]. La RA requiere de componentes como el hardware: computadora o un dispositivo móvil, una pantalla, una cámara y un marcador (geolocalización, reconocimiento de imágenes) y el software siendo frameworks de RA, en dispositivos móviles con Android se puede utilizar Vuforia, ARToolKit entre otros. La RA se puede presentar de dos formas: reconociendo una imagen marcadora o mediante un punto de localización geográfica, por eso al señalar se requiere que un marcador brinde las dos opciones. Cuando se utiliza un marcador, básicamente se asocia un modelo virtual en tercera dimensión a un objeto físico; cuando se usa la localización, en lugar de reconocer un marcador, se asigna información digital a un grupo de coordenadas geográficas.

Los procesos que se realiza la RA por medio de un marcador comienza por una cámara que muestra una señal de video en tiempo real, la señal es digitalizada e interpretada por el programa que, a su vez, identifica el marcador y lo asocia con el contenido digital asignado a él y, finalmente, el contenido digital es reproducido dentro del marco de la señal de video a través de la pantalla del dispositivo móvil o el monitor de la computadora [10]. En este escenario, la realidad aumentada puede ayudar a los estudiantes a contextualizar su aprendizaje, ya que la creación de contenidos altamente interactivos basados en realidad aumentada apoya el proceso de aprendizaje, donde se utilizó el reconocimiento de una imagen marcadora para los contenidos interactivos.

3. Análisis y diseño del prototipo

Para el diseño se utilizó el Diseño Centrado en el usuario (DCU) y la metodología de prototipos Se realizaron entrevistas para la definición de los requerimientos de los escenarios de aprendizaje en el área de biología del “Sistema circulatorio cardiovascular”, enfocándose en apoyar el desarrollo de las competencias disciplinares, las cuales son: 1) Identifica y localiza los principales órganos que componen el aparato circulatorio, 2) Analiza su funcionamiento básico, 3) Reconoce algunas enfermedades comunes del aparato circulatorio y de la sangre y 4) Detecta como prevenirlas.

Para el análisis y el diseño del sistema se determinaron los Casos de Uso del sistema en UML. El sistema permite identificar dos usuarios, son:

- Usuario Docente: Puede realizar consultas generales del seguimiento de estudiantes con respecto a los contenidos, actividades y evaluación del tema.
- Usuario Alumno con Moderada/Severa: Consulta los contenidos y materiales de información, al realizar el diagnóstico, el sistema adapta las interfaces y presenta los contenidos que requiere aprender y propone una serie de actividades lúdicas para el desarrollo de las competencias del sistema circulatorio cardiovascular (ver figura 1).

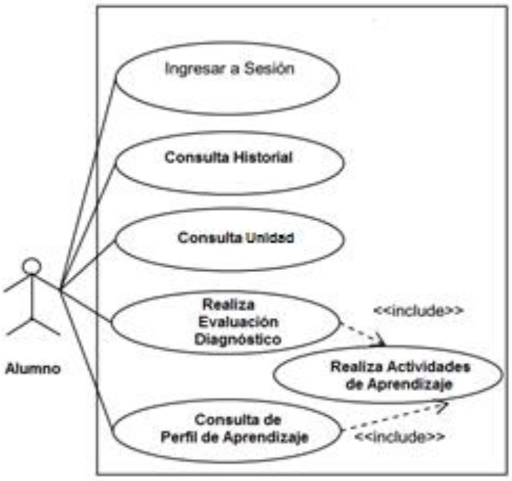

Ingresar a Sesión
Consulta Historial
Consulta Unidad
Realiza Evaluación Diagnóstico
<<include>>
Realiza Actividades de Aprendizaje
Consulta de Perfil de Aprendizaje
<<include>>
Alumno

Fig. 1. Diagrama de caso de uso del alumno

Así también, se definieron los storyboards, el diseño de interfaces de usuario donde se muestra los temas de cada unidad, los elementos de apoyo (imágenes), en esta sección se podrá navegar con el Kinect y la detección de la mano para seleccionar una imagen o texto, como se muestra en la Figura 2.

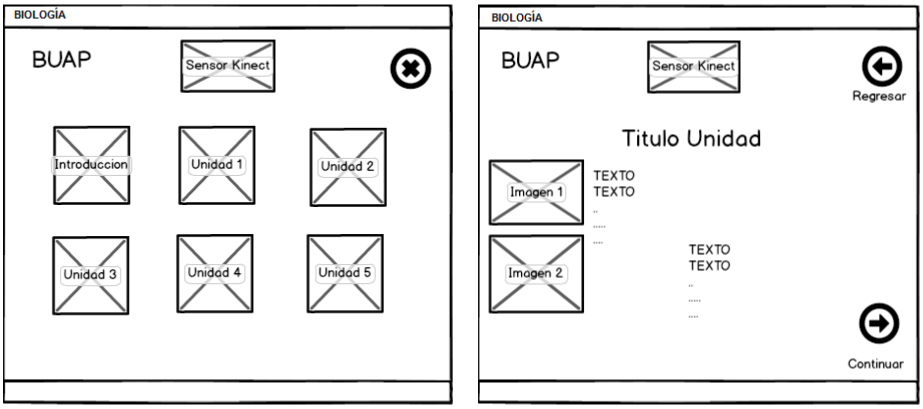
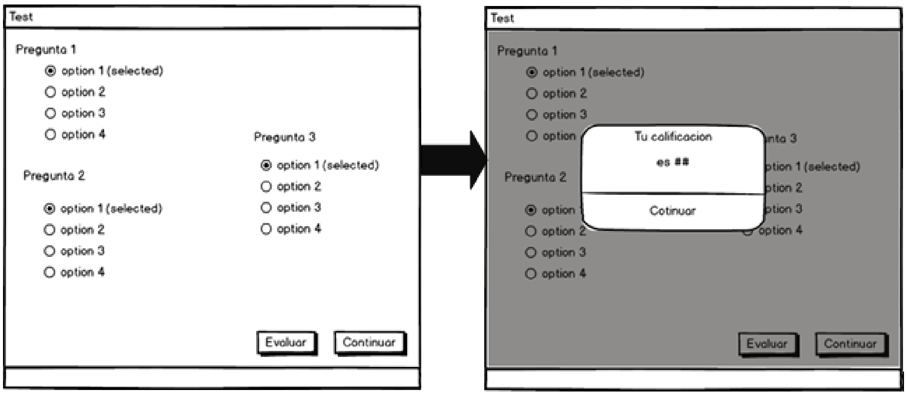


Fig. 2. Diseño de unidades del sistema

En la Figura 3, muestra el diseño de pantalla del Test por unidad la cual contendrá cuatro preguntas que se evaluaran y al terminar se presenta una ventana (modal) con el resultado obtenido, teniendo el formato de exámenes del Programa para la Evaluación Internacional de los Alumnos (PISA) [11]. Las preguntas de ciencias de PISA intentan evaluar hasta qué punto los alumnos aplican algún tipo de pensamiento científico a las situaciones que puedan encontrarse en sus vidas cotidianas y se distinguen tres dimensiones para la evaluación de las ciencias: conceptos, procesos y situaciones.



Test

Pregunta 1

● option 1 (selected)
○ option 2
○ option 3
○ option 4

Pregunta 2

● option 1 (selected)
○ option 2
○ option 3
○ option 4

Pregunta 3

● option 1 (selected)
○ option 2
○ option 3
○ option 4

Evaluar | Continuar

Tu calificacion
es ##

Cotinuar

Fig. 3. Vista de evaluación de test formato PISA

Para la arquitectura del sistema, se identificaron elementos para el reconocimiento de de movimientos corporales mediante el Kinect, se elaboraron archivos y/o scripts en C++ y C# [11]. Por otra parte, Vuforia es un SDK para dispositivos móviles que permite la creación de aplicaciones RA, proporciona (API) en C++ y Java, es compatible con el desarrollo nativo para iOS y Android, a la vez permitiendo el desarrollo de aplicaciones de RA, fáciles de transportar a ambas plataformas y el manejo compatible con dispositivos móviles como el iPhone, iPad, teléfonos y tabletas con una conexión a la base de datos MySql para la implementación y control de los usuarios e imágenes.

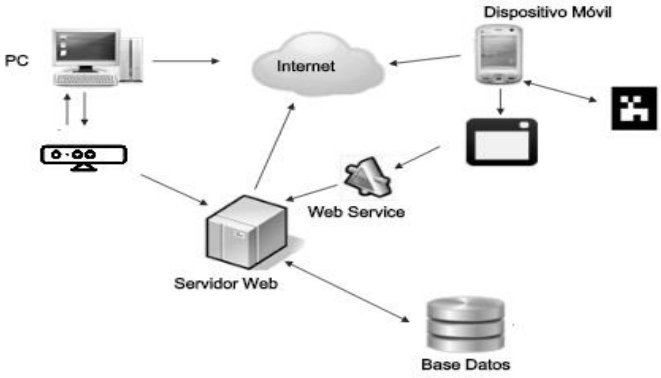


Fig. 4. Arquitectura del sistema.

4. Desarrollo y pruebas del sistema

El desarrollo del sistema Sim’s Anatomy con el dispositivo Kinect® se programó en Greenhouse SDK en plataformas iOS, en C++ que permite la interacción a través de la entrada de reconocimiento de movimientos, como se muestra en la Figura 5. La librería G-speak™ permitió el manejo de múltiples pantallas de dispositivos móviles y en red. Su forma de trabajar se asemeja a una estructura cliente-servidor, la conexión se hace de manera local, mediante la dirección IP de la computadora. En cuanto al servidor (elegido); es necesario tener dos procesos ejecutandose: pool-tcp-server y pool-server-zeroconf-adapter, presentando extensiones a otros dispositivos.

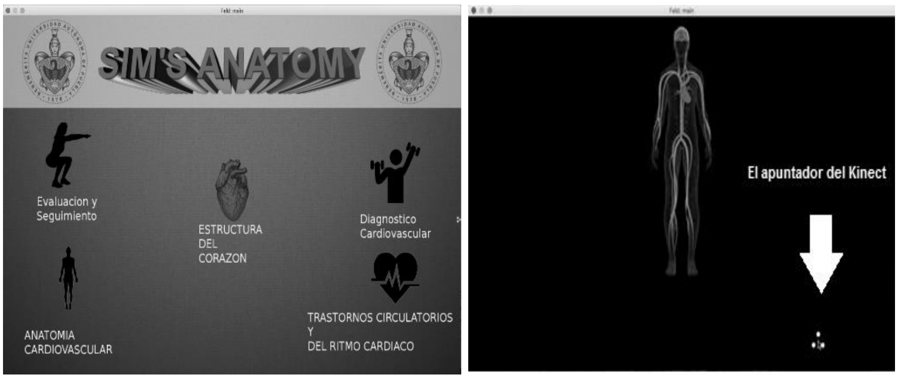


Fig. 5. Pantallas del sistema con uso del kinect.

Los estudiantes con discapacidad visual utilizan otros sentidos para la percepción, como el tacto y la vista, lo que fue considerado en el sistema para estimular estos dos sentidos e incluso el auditivo dependiendo del tipo de discapacidad auditiva del usuario. Por lo cual, el sistema se diseñó para lograr una interacción basada en gestos (movimientos o posicionamiento) y representación visual logrando una mayor interacción con el sistema y proporcionando una estimulación física y cognitiva para el desarrollo de las competencias de los estudiantes.

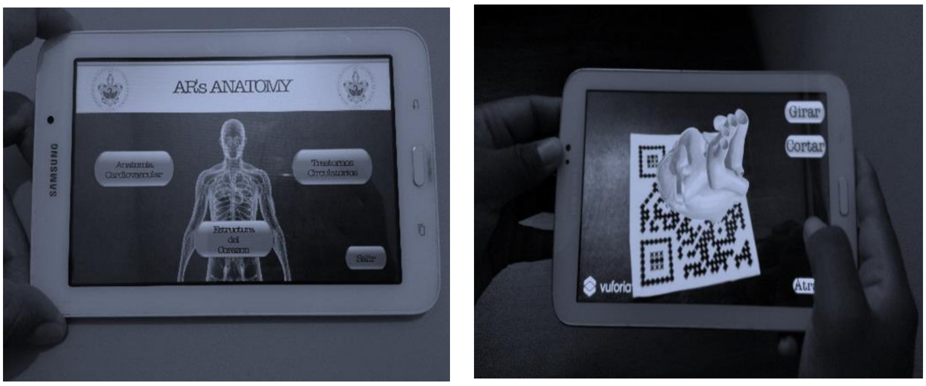


Fig. 6. RA del sistema circulatoria en Tablet con Android.

Para los contenidos enriquecidos con RA, versión dispositivo móvil la extensión del sistema conocida como AR's Anatomy se desarrolló con Vuforia en Unity y C++ con

Android. Las actividades de exploración mediante RA, basada en marcadores y en un sistema de coordenadas, permitió contenidos interactivos en 3D, ver Figura 6.

El sistema considera tres niveles de desempeño de las competencias, la cuales son: 1) Iniciada, 2) En Proceso y 3) Desarrollada. De acuerdo al perfil pueden ser: Principiante, Intermedio y Avanzado. Para evaluar los contenidos se maneja en tres puntajes: 1) Bajo, menor a la media, 2) Regular, dentro de la media y 3) Alto, superior a la media, lo cual permite llevar el seguimiento académico del proceso de aprendizaje. El prototipo del sistema fue piloteado con un grupo focal de 6 estudiantes con discapacidad auditiva, de moderada-sordera total, los cuales pertenecen a educación media superior.

Con respecto a las pruebas de funcionalidad se aplicó la técnica de inspección con la experiencia de usuario. Para lo cual se presentaron tres posibles escenarios para los usuarios finales, en el uso de las interfaces naturales con el Kinect y con el dispositivo móvil con RA que a continuación se describen:

- Situación 1: Al estudiante con discapacidad moderada se le dio una breve explicación del uso del sistema y solo se le acompañó al inicio de la actividad.
- Situación 2: Al estudiante con discapacidad profunda se le explicó el uso del sistema y se le acompañó en la realización de algunas actividades.
- Situación 3: Al estudiante con discapacidad sordera total se le explicó el uso del sistema y se le acompañó en la realización de todas las actividades.

Para cada una de las situaciones los usuarios debieron cumplir ciertas tareas, con la finalidad de comprobar el funcionamiento del uso del sistema.

- Tarea 1: Poder ser detectado por el dispositivo del Kinect® y registrar su perfil.
- Tarea 2: Encontrar el menú de navegación y seleccionar una opción.
- Tarea 3: Utilizar los escenarios de aprendizaje y recorrerlos mediante el dispositivo natural del Kinect®, comandos de señas y movimientos corporales.
- Tarea 4: Activar el dispositivo móvil e interactuar con el sistema, seleccionando contenidos de RA.
- Tarea 5: Realizar las autoevaluaciones (diagnósticos) y reconocer su nivel de competencia.

Después de realizar las pruebas y de haber asignados solo a dos estudiantes por cada situación, se observó que los estudiantes con una breve explicación y acompañamiento al inicio (Situación 1), el desempeño fue en promedio del 94.8% del cumplimiento de las tareas, mientras que los usuarios que se les apoyo en algunas actividades (Situación 2) fue del 90%, y para los estudiantes con sordera total (Situación 3), lograron realizar las tareas en un 92.8%, lo cual representa una interfaz natural intuitiva, sencilla y agradable para los estudiantes, mientras los contenidos enriquecidos en dispositivos móviles les facilita el acceso a la información, interacción y motivación . Por otra parte, el desarrollo de las competencias se enfatiza en las competencias 1 y 2, que se muestra en la Tabla 2.

Finalmente, se aplicó una encuesta de satisfacción, Valoración del Software [12], la cual evalúa siete criterios: Navegación, Interactividad, Inmersión, Usabilidad, Creatividad, Efectividad y Calidad, con una escala de 1 a 5, cuyo promedio obtenido fue de 4.6 (92%), lo que nos indica que la satisfacción del estudiante es alta para poder desarrollar sus competencias y un aprendizaje significativo, debido al uso de las interfaces naturales y de la RA como se muestra en la siguiente Figura 7.

Table 2. Nivel de las competencias

Competencia	Iniciada Bajo	En Proceso Regular	Desarrollada Alto
Competencia$_1$	12%	56.5%	31.5%
Competencia$_2$	15%	53%	32%
Competencia$_3$	8%	62%	30%
Competencia$_4$	16.6%	63.4%	20%

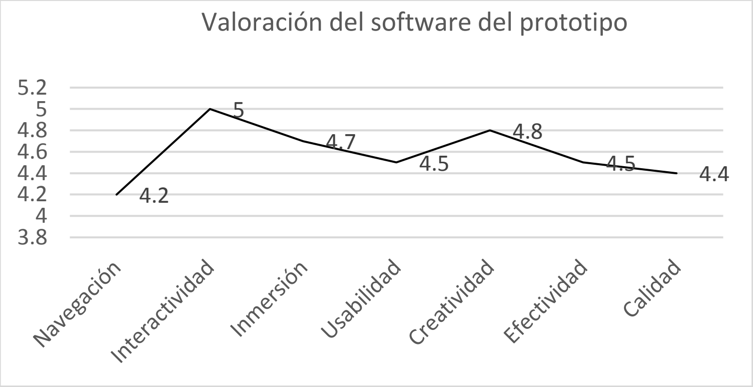


Fig. 7. Resultados de satisfacción del Prototipo

5. Conclusiones y trabajos futuros

Una de las principales contribuciones del sistema como prototipo, es la aportación a la inclusión de los estudiantes con discapacidad auditiva, propiciando experiencias de aprendizaje significativas apoyadas del uso de interfaces naturales y realidad aumentada, lo que permite el proceso de adquisición y desarrollo de las competencias disciplinares de la materia de biología y fortalecer el perfil de los estudiantes con alguna discapacidad auditiva.

El objetivo de integrar realidad aumentada a contenidos interactivos, ha permitido proponer que las nuevas tecnologías aplicadas en la educación y en otras áreas, es una opción que se debe implementar en el diseño y desarrollo de materiales educativos interactivos para estudiantes y ayudar al mismo tiempo el desarrollo de habilidades digitales en distintas plataformas y dispositivos. La perspectiva de este trabajo es realizar la evaluación de la accesibilidad y usabilidad del sistema propuesto ampliando la muestra de estudio y poder proveer a los alumnos con discapacidad auditiva materiales y herramientas digitales que puedan enriquecer el proceso de aprendizaje en distintas asignaturas logrando adaptaciones a exámenes, materiales, herramientas de estudio para apoyar la vida académica y promover la inclusión de los estudiantes con alguna discapacidad en el uso de las tecnologías digitales logrando disminuir los limitantes y barreras de la brecha digital que se presentan al interior de las instituciones educativas.

Referencias

1. Organización de las Naciones Unidas para la Educación, La Ciencia y la Cultura (UNESCO): Educación para Todos en las Américas. Marco de Acción Regional, República Dominicana (2000)
2. Organización Mundial de la Salud (OMS): Resumen de Informe Mundial de Discapacidad, (2001)
3. Villalba, A.: La pérdida de audición. Tipos de sordera y consecuencias que se derivan para la educación, Consellería de Cultura, Educación y Ciencia. Generalitat, Valenciana (1996)
4. Valli, A.: Notes on natural interaction (2005)
5. Anki, D., Yogesh, B., Abin, A., Rekha, S: Project MUDRA: Personalization of Computers using Natural Interface. International Journal of Computer Applications, Volume 54, Number 17, pp. 42–46 (2012)
6. Clark, R., Pua, Y., Fortin, K., Ritchie, C., Webster, K., Denehy, L., Bryant, A. L.: Validity of the Microsoft Kinect for assessment of postural control. Gait and Posture, Volume 36, Issue 3, pp. 372–377 (2012)
7. Chang, Y., Chen, S., Huang, J.: A Kinect-based system for physical rehabilitation: A pilot study for young adults with motor disabilities. Research in Developmental Disabilities, Volume 32, Issue 6, pp. 2566–2570 (2011)
8. Herrera, M.: Las Nuevas Tecnologías en el Aprendizaje Constructivo. Revista Iberoamericana de Educación (2004)
9. Johnson, L., Adams, S., Gago, D., García, E., Martín, S.: NMC perspectivas tecnológicas: educación superior en América Latina 2013-2018. Un análisis regional del informe Horizon del NMC. Austin, Texas: The New Media Consortium (2013)
10. Badilla, M., Sandoval A.: Realidad Aumentada como Tecnología Aplicada a la Educación Superior: Una Experiencia en Desarrollo. Universidad Estatal a Distancia (2016)
11. Microsoft Kinect Sensor (2015)
12. Acuña, A., Romo, M.: Diseño Instruccional Multimedia. Pearson Education, México (2011)

Impreso en los Talleres Gráficos
de la Dirección de Publicaciones
del Instituto Politécnico Nacional
Tresguerras 27, Centro Histórico, México, D.F.
noviembre de 2017
Printing 500 / Edición 500 ejemplares