Advances in Natural Language Processing and Computational Linguistics

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 115,** septiembre 2016. Tiraje: 500 ejemplares. *Certificado de Reserva de Derechos al Uso Exclusivo del Título* No. : 04-2005-121611550100-102, expedido por el Instituto Nacional de Derecho de Autor. *Certificado de Licitud de Título* No. 12897, *Certificado de licitud de Contenido* No. 10470, expedidos por la Comisión Calificadora de Publicaciones y Revistas Ilustradas. El contenido de los artículos es responsabilidad exclusiva de sus respectivos autores. 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 115**, September 2016. Printing 500. The authors are responsible for the contents of their articles. Printed in Mexico City, in the IPN Graphic Workshop – Publication Office.

Volume 115

Advances in Natural Language Processing and Computational Linguistics

Noé Alejandro Castro-Sánchez (ed.)

Instituto Politécnico Nacional, Centro de Investigación en Computación
México 2016

ISSN: 1870-4069

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

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

Indexed in LATINDEX, DBLP and Periodica

Printing: 500

Printed in Mexico

Editorial

La integración de la tecnología en la sociedad actual y su uso en la resolución de problemas de la vida cotidiana, prioriza la necesidad de alcanzar una comunicación natural entre personas y computadoras. Esto ha venido motivando el desarrollo de investigación y de aplicaciones concernientes a la Inteligencia Artificial, y en particular de la disciplina conocida como Procesamiento de Lenguaje Natural.

Este volumen está conformado por 15 artículos seleccionados a partir de un estricto proceso de arbitraje por pares de rigor internacional, en donde se tomó en cuenta la originalidad, aportación y calidad técnica de los mismos.

El material que se presenta en este compendio trata temas relacionados con tendencias actuales de investigación, en donde destacan el perfilado de autores de textos cortos, el análisis de sentimientos, la construcción de tesauros y lexicones, la detección de malware, la identificación del idioma a través de información acústica, la traducción del español al Lenguaje de Señas Mexicano y la expansión y autocompletado de consultas, entre otros.

Estoy seguro que estos trabajos capturarán la atención de estudiantes e investigadores de las Ciencias de la computación en general, y en particular de quienes se ven interesados en el estudio y modelado de los diversos fenómenos del lenguaje a través de medios computacionales.

Extiendo mi agradecimiento a la Sociedad Mexicana de Inteligencia Artificial (SMIA), y al Instituto Tecnológico de Cancún por su invaluable apoyo para la preparación de este volumen de la revista Research in Computing Science.

El proceso para someter, revisar y elegir los artículos se realizó desde la plataforma libre EasyChair (easychair.com).

Noé Alejandro Castro Sánchez
Editor Invitado
CENIDET, México

Septiembre 2016

Table of Contents

Page

Detección de malware con modelo de lenguaje y su clasificación mediante SVM

Alex I. Valencia-Valencia[1], Sofía N. Galicia-Haro[2]

[1] UNAM, Posgrado en Ciencias e Ingeniería de la Computación, México

[2] UNAM, Facultad de Ciencias, México

letras_vivas@comunidad.unam.mx, sngh@fciencias.unam.mx

Resumen. La detección de malware representa una tarea cada vez más compleja, debido a las avanzadas técnicas de evasión a la detección; empleadas por los desarrolladores de malcode: desde el ofuscado de código, uso de polimorfismo, hasta variantes de malware que destruyen el disco duro si detectan que están siendo analizados. En la presente investigación se realiza un análisis dinámico de seis tipos de malware: Troyanos, Gusanos, Virus, Troyanos Espía, Puertas Traseras y Rootkits, además de un conjunto de Whiteware empleando el modelo de lenguaje de n-gramas en las llamadas al sistema del tipo WinAPI para cada muestra. Finalmente utilizamos el método de Máquinas de Soporte Vectorial (SVM) con kernel polinomial como algoritmo de aprendizaje automatizado para predecir la clasificación de malware, en nuestro caso con un rendimiento promedio de 70% y de 100% para el Whiteware, lo cual nos permite concluir que el modelo determina si la muestra es maliciosa y de esta manera puede llevar a cabo la detección de malware.

Palabras clave: Detección de malware, clasificación de malware, N-grama, SVM.

Language Model for Malware Detection and Classification based on SVM

Abstract. Malware analysis is a more difficult task each time, due to malware developers are taking care of avoiding and detection techniques, from obfuscation to destroying, hard drive (if malware detect that has been analyzed). In this paper we release a malware dynamic analysis of six types of malware: Trojans, Worms, Virus, Trojan-Spys, Backdoors y Rootkits, moreover a set of Whiteware using an n-grams language model in Win API calls for every sample. Finally, we used a Polynomial Kernel SVM to malware classification prediction, where we have obtained 70% of performance and malware classification and 100% for Whiteware classification in which case determine if is a malicious sample and in this form we can do malware detection.

Keywords: Malware detection, malware classification, N-gram, SVM.

1. Introducción

Al día de hoy las computadoras se han convertido en grandes herramientas de procesamiento y en este sentido los códigos maliciosos han evolucionado tanto en el daño causado cómo en las características que permiten ocultarlos del análisis. Respecto a las estadísticas la firma Symantec en su reporte de Amenazas de Seguridad en Internet de 2015 manifiesta que en el 2014 fueron introducidas 317 millones de variantes de malware. Mantenerse al día con la gran cantidad de variantes es desalentador para las organizaciones [1]. El método de análisis de malware es uno de los problemas clave en la técnica de detección de intrusos. En la literatura, los principales métodos de análisis de malware se basan en análisis de contenido estático y en el comportamiento dinámico [2]. En el análisis estático, las características se extraen del código binario de los programas y se usan para crear modelos que los describan. Los modelos se usan para distinguir entre malware y software legítimo [3]. Sin embargo el análisis estático falla en diferentes técnicas de ofuscación de código que se usa por los desarrolladores de códigos maliciosos y también en malcodes metamórficos y polimórficos [4].

En las técnicas de ofuscación el código fuente y el código binario se transforman de tal manera que tanto el proceso de decompilación sea más difícil como la lectura y análisis del mismo. Aunado a lo anterior el polimorfismo de malware consiste en cambiar la apariencia del malware a través de métodos de cifrado, de agregación de datos o eliminación de datos [5]. Debido a que el análisis dinámico de malware consiste en el estudio del mismo a través de su ejecución en un entorno controlado, su principal ventaja es que dicho análisis de comportamiento no puede ser ofuscado [6-7]. Sin embargo, hay algunas limitaciones en el análisis dinámico. Ya que cada muestra de malware debe ejecutarse dentro de un entorno seguro por un tiempo específico para monitorear el comportamiento. El proceso de monitoreo consume tiempo y debe asegurarse que la ejecución del malware no infecta la plataforma [8]. Los entornos seguros discrepan sólo un poco de un entorno de ejecución real y el malware puede comportarse de diferente manera en los dos entornos, causando una bitácora inexacta del comportamiento del malware [9]. Además de que algunas acciones del software malicioso se activan o ejecutan bajo ciertas condiciones (la fecha y hora del sistema o alguna entrada particular proporcionada por el usuario) puede no detectarse por el entorno virtual seguro [10].

Sin embargo, el análisis dinámico es un complemento necesario al enfoque estático como una medida preventiva de ofuscación de código. El análisis dinámico de malware basado en comportamiento consiste de una colección de muestras de malware, la ejecución de dichas muestras, entornos de monitoreo y la determinación de un modelo de características de comportamiento y análisis de comportamiento (clustering, clasificación, reconocimiento, etc.). La investigación de recolección de malware, ejecución de malware y métodos de monitoreo ha alcanzado un resultado maduro. Algunos sistemas de recolección y métodos se proponen basados en honeypot, como: Dionaea [11] y Kippo [12]. Los sistemas típicos tales como Anubis [13-14], CWSandbox [15], CuckooSandBox [16] ejecutan el malware en un ambiente controlado y monitorean el comportamiento del mismo (dicho método se nombra Sand-boxing), finalmente genera un reporte de comportamiento para cada muestra. El análisis profundo es necesario para revelar características del comportamiento y la detección de malware. Dos principales conceptos para el análisis automático de comportamiento se

han propuesto: clustering y clasificación [17-18]. A diferencia de muchos artículos en la literatura, que realizan el análisis de comportamiento con reportes de syscalls utilizando modelos de secuencias y/o los resultados del análisis estático realizados por la herramienta SandBox [4], [7], [13-19], en la presente investigación usamos las llamadas al sistema del tipo WinAPI y un modelo de lenguaje basado en n-gramas, para que el modelo considere la detección de malware tomamos en cuenta un conjunto de Whiteware o "malware benigno" con programas en la carpeta System32.

El artículo se organiza como sigue: en la segunda sección presentamos los elementos que se utilizan como datos para nuestra base de conocimiento la cual consiste de los 5-gramas de Win API calls de las muestras de malware. En la tercera sección describimos los parámetros con que se realizó el proceso de aprendizaje automatizado a través de SVM en Weka. En la cuarta sección primero describimos el comportamiento de los diferentes tipos de malware que se utilizaron en el experimento y posteriormente presentamos las estadísticas de los 5-gramas que se emplearon para el método de clasificación. En la quinta sección damos los resultados correspondientes obtenidos por Weka al realizar 10 iteraciones de la generación del modelo por medio de SVM.

Finalmente presentamos nuestras conclusiones y planteamos las líneas de trabajo futuro.

2. N-gramas en API Windows Calls

En los campos de probabilidad y lingüística computacional, un n-grama es una secuencia contigua de n elementos de una determinada secuencia de texto o de habla. Los elementos pueden ser fonemas, silabas, letras de acuerdo a su aplicación. Los n-gramas típicamente se recolectan de un corpus de texto o de habla. Cuando los elementos son palabras los n-gramas también pueden ser llamadas "shingles" [20].

Un n-grama de tamaño 1 se nombra "unigrama", el de tamaño 2 "bigrama", 3 "trigrama". Los n-gramas de mayor tamaño son nombrados por el valor de n, es decir; "4-grama", "5-grama", etc.

La siguiente tabla muestra diferentes ejemplos de secuencias y su modelo de secuencia de n-grama presentada en [21]:

Tabla 1. Ejemplos de n-gramas.

Campo	Unidad	Ejemplo de secuencia	Secuencia 1-grama	Secuencia 2-grama	Secuencia 3-grama
Secuencia de Proteinas	Amino ácido	…Cys-Gly-Leu-Ser-Trp …	…, Cys, Gly, Leu, Ser, Trp, …	…, Cys-Gly, Gly-Leu, Leu-Ser, Ser-Trp, …	…, Cys-Gly-Leu, Gly-Leu-Ser, Leu-Ser-Trp, …
Secuencia DNA	Pares base	…AGCTTCGA…	…, A, G, C, T, T, C, G, A, …	…, AG, GC, CT, TT, TC, CG, GA, …	…, AGC, GCT, CTT, TTC, TCG, CGA, …

Para el objetivo de este experimento utilizamos un análisis de 5-gramas en el reporte de WinAPI calls, el cual se llevó a cabo por la SandBox Cuckoo [16] dicho reporte está relacionado con los resultados del monitor de procesos de Windows, el cual se ejecuta durante el análisis de cada muestra de malware.

La API de Windows también llamada WinAPI es el conjunto núcleo de las interfaces de programación de aplicaciones (APIs) de Microsoft disponibles en los sistemas operativos de Microsoft Windows. Con dichas APIs de Windows pueden desarrollarse aplicaciones que se ejecuten exitosamente en todas las versiones de Windows mientras se aprovecha de las características y capacidades únicas de cada versión. Notar que esta fue formalmente llamada la API Win32. El nombre de Windows API refleja de manera más precisa su raíz en Windows de 16-bit y su soporte en Windows de 64-bit.

La siguiente lista es una referencia de contenido para la API de Windows de 32-bit tanto para aplicaciones de escritorio como de servidores, y dentro de cada categoría se encuentran las Win API calls que pudieron ser ejecutadas por cada muestra de malware [22]:

1. User Interface
2. Windows Environment (Shell)
3. User Input and Messaging
4. Data access and storage
5. Diagnostics
6. Graphics and Multimedia
7. Devices
8. System Services
9. Security and Identity
10. Application Installation and Servicing
11. System Admin and Management
12. Networking and Internet
13. Deprecated or legacy APIs

En la siguiente tabla se muestran ejemplos de los 5-gramas obtenidos de las muestras de malware:

Tabla 2. Ejemplos de 5-gramas de Win API calls obtenidos.

No.	5-grama de Win API Call
1	{NtOpenFile,NtCreateSection,NtCreateFile,NtQueryInformationFile,NtSetInformationFile}
2	{LdrGetProcedureAddress,NtQueryInformationFile,GetSystemMetrics,NtCreateSection,ZwMapViewOfSection}
3	{LdrGetProcedureAddress,LdrGetProcedureAddress,LdrGetProcedureAddress,NtFreeVirtualMemory,GetSystemMetrics}
4	{OpenServiceW,RegQueryValueExA,CreateThread,RegCloseKey,RegCloseKey}
5	{NtDelayExecution,ZwMapViewOfSection,DeleteFileA,NtDelayExecution,DeleteFileA}

3. Máquinas de soporte vectorial

Las Máquinas de Soporte Vectorial son un conjunto de algoritmos de aprendizaje supervisado desarrollados por Vladimir Vapnik y su equipo en los laboratorios AT&T. Estas construyen un hiperplano o conjunto de hiperplanos en un espacio de dimensionalidad muy alta (o incluso infinita) que puede ser utilizado en problemas de clasificación o regresión. Una buena separación entre las clases permitirá una clasificación correcta [23]. Dicho lo anterior y al obtener la similitud vectorial entre cada registro de la tabla binaria generada con los 5-gramas se determinó que sólo la clase Whiteware era la única diferente a todas las clases. Para llevar a cabo el experimento se utilizó el software Weka en su versión 3.7.13 y se realizaron 10 separaciones aleatorias de datos de entrenamiento y datos de prueba, además de los siguientes parámetros del clasificador SVM:

Tipo de experimento: División porcentual de conjunto de entrenamiento y prueba (con datos aleatorizados).
Porcentaje de entrenamiento: 70
Clasificador: Función SMO
Tamaño del lote: 1124
C: 1.0
Épsilon: 1.0E-12
Tipo de Filtro: Normalizar los datos de entrenamiento
Kernel: PolyKernel -E 1.0 -C 250007
Número de cifras decimales: 6

4. Diseño del experimento

En el experimento descargamos las 900 muestras de los 6 diferentes tipos de malware de malwr.com [24] considerando diferentes familias de cada tipo [25]. Dichos tipos de malware corresponden a los siguientes:

1. **Puertas traseras o Backdoors:** son un método externo en el proceso de autenticación o en otros controles de seguridad con el fin de acceder a un sistema de cómputo o a los datos contenidos en el mismo [8].
2. **Troyanos:** es un tipo de software malicioso que se empaqueta junto con una pieza útil del software o se hace pasar por una pieza de software útil. Una vez que el troyano es activado, que por lo general pasa desapercibido por el usuario, se libera una carga útil de ellos ya sea como un virus o una puerta trasera que puede permitir a un usuario acceder remotamente al sistema [6].
3. **Troyanos Espía:** software que obtiene información de una persona u organización sin su conocimiento y que puede enviar tal información a otra entidad sin el consentimiento del cliente, o que impone el control sobre una computadora sin el conocimiento del cliente [3].
4. **Gusanos:** Un programa usualmente pequeño que auto replica su contenido en sí mismo y que invade computadoras en una red y generalmente realiza acciones destructivas [10].

5. **Rootkits:** programas maliciosos que se ocultan en el sistema a través de modificaciones en las herramientas del sistema, filtrando activamente información de estado del sistema de los usuarios, enmascarando la presencia de archivos, servicios y canales de comunicación maliciosos [9].
6. **Virus:** Un programa que está usualmente oculto dentro de otro programa aparentemente inocuo y que produce copias de sí mismo e inserta otros programas y usualmente realiza una acción maliciosa (como destruir datos) [7].

Como parte del pre procesamiento de los datos se obtuvieron los 5-gramas por cada muestra de los seis tipos de malware incluyendo el whiteware consiguiendo un total de 85,013 5-gramas tal cómo se muestra en la Tabla 3, de los cuales 63,204 son únicos, y con estos últimos se generó una tabla binaria en representación de la presencia de cada 5-grama para cada muestra de clase, dicha tabla sirvió como entrada para el algoritmo de máquinas de soporte vectorial cuyos resultados se presentan en el siguiente punto. Para los diferentes tipos de malware se utilizaron 150 muestras de diferentes variantes del mismo y para Whiteware se utilizaron 224 muestras. Resultando una tabla de 63205x1124 campos (incluyendo la variable categórica de clase).

Tabla 3. Número de 5-gramas por cada tipo de malware.

Tipo de malware	Número de 5-gramas
Backdoors	6,343
Troyanos	15,378
Troyanos-Espía	11,621
Gusanos	33,888
Rootkits	606
Virus	12,212
Whiteware	4,965
Total	85,013

5. Resultados

Al realizar diez iteraciones con diferentes algoritmos de clasificación mostrados en la Tabla 4, puede apreciarse que las SVM con Kernel Polinomial se posicionan en el segundo mejor promedio de sensibilidad después de Random Forest y su tiempo de construcción del modelo es menor que este.

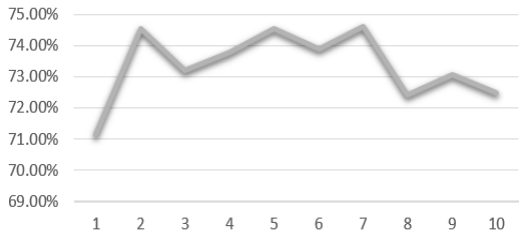

Fig. 1. Rendimiento del clasificador SVM.

Una vez determinado el algoritmo de Máquinas de Soporte Vectorial con Kernel Polinomial se ejecutó con los parámetros ya mencionados con el software Weka, sobre la misma base de conocimiento con los seis tipos de malware y whiteware, el rendimiento promedio fue de 70.16%, alcanzando un máximo de 75.65% y un mínimo de 71.13%.

Tabla 4. Tabla de comparación de sensibilidad entre algoritmos por tipo de malware.

	Logit Boost	**Naive Bayes Multinomial**	**Random Forest**	**SVM RBF Kernel**	**SVM Poly Kernel**	**SVM Poly Kernel Normalized**
Backdoor	44.2%	55.8%	58.1%	53.5%	67.4%	58.1%
Trojan	47.2%	44.4%	55.6%	50.0%	50.0%	55.6%
Trojan-Spy	72.1%	69.8%	72.1%	53.5%	67.4%	74.4%
Worm	37.0%	57.4%	57.4%	50.0%	48.1%	48.1%
Rootkit	93.2%	97.7%	90.9%	93.2%	95.5%	95.5%
Virus	44.2%	48.1%	65.4%	50.0%	57.7%	51.9%
Whiteware	100.0%	100.0%	100.0%	100.0%	100.0%	100.0%
Promedio	64.1%	69.4%	73.0%	66.2%	70.9%	70.1%
*Tiempo	251.02	0.85	91.12	56.46	57.73	61.28

*Tiempo significa Tiempo de construcción del modelo en segundos.

Es importante mencionar que la distribución promedio en el conjunto de prueba fue la siguiente: Backdoors: 42, Troyanos: 41, Troyanos Espía: 39, Gusanos: 46, Rootkits: 54 Virus: 45 y Whiteware: 65. Con dicha información se obtuvo la siguiente Tabla de resultados por clase, con el mismo número de iteraciones ya mencionadas, dónde puede apreciarse que el mejor rendimiento fue obtenido al clasificar Rootkits además del Whiteware y en último lugar los Gusanos.

Tabla 5. Rendimiento del clasificador por clase.

Clase	**TP**	**FP**	**Prec**	**F-M**	**MCC**	**ROC**	**PRC**
Backdoor	67%	5%	64%	66%	61%	87%	54%
Troyano	50%	6%	49%	49%	43%	83%	37%
Troyano Espía	67%	2%	81%	73%	70%	90%	65%
Gusano	48%	5%	65%	55%	49%	75%	43%
Rootkit	96%	3%	82%	88%	87%	97%	81%
Virus	58%	6%	64%	61%	54%	82%	47%
Whiteware	100%	6%	80%	89%	87%	97%	80%
Promedio	79%	5%	70%	70%	66%	87%	59%

Dónde: TP significa Tasa de Verdaderos Positivos, FP – Tasa de Falsos Positivos, Prec – Precisión, F-M - Valor-F, MCC – Coeficiente de Correlación de Matthews, ROC - Área ROC y PRC - Área PRC.

Como parte de la investigación se obtuvo la similitud vectorial entre cada registro de la tabla (similitud en términos de presencia de 5-gramas por muestra) utilizando la medida del coseno [26] algunos de los resultados se muestran en la Tabla 6.

Tabla 6. Tabla de porcentajes de similitud vectorial entre clases.

	Backdoor	**Troyano**	**Troyano Espía**	**Gusano**	**Rootkit**	**Virus**	**Whiteware**
Backdoor	20.7%	11.4%	22.9%	10.5%	18.2%	16.6 %	0%
Troyano	11.4%	9.5%	13.4%	8%	12.3%	10.1 %	0%
Troyano Espía	22.9%	13.4%	29.8%	12.5%	22.5%	19.6 %	0%
Gusano	10.5%	8%	12.5%	9%	0%	0%	0%
Rootkit	18.2%	12.3%	22.5%	0%	19.4%	0%	0%
Virus	16.6%	10.1%	19.6%	0%	0%	13.3 %	0%
Whiteware	0%	0%	0%	0%	0%	0%	1%
Promedio	14.3%	7.6%	12%	1.2%	10.3%	1.9%	0

6. Conclusiones y trabajo futuro

Cómo puede apreciarse en la gráfica, el modelo tiene un rendimiento que logra mantenerse en el 70%, y además cabe mencionar que el nivel de contribución de 5-gramas para el llenado de la tabla por cada tipo de malware tiene correlación con el rendimiento de su clasificación y ésta es inversamente proporcional; siendo que de los 85,013 5-gramas (incluyendo las repeticiones) los Gusanos tienen un mayor número de 5-gramas, es decir, la entropía de información proporcionada por su número de 5-gramas es muy alta y esto se muestra en la Tabla 6 que comparte similitud con dos tipos de malware, ergo tienen procesos con mayor cantidad de Win API calls lo cual hace más difícil su detección con este enfoque y análogamente los Rootkits con tan sólo 606 5-gramas nos dice que la variación entre cada muestra de este tipo es pequeña respecto a sus 5-gramas lo cual se manifiesta en la Tabla 6 ya que las muestras de este tipo comparten el tercer mayor coeficiente de similitud y por tanto más fácil de detectarse cómo puede apreciarse en la Tabla 4. Adicionalmente respecto a trabajos relacionados el costo computacional que representa obtener los 5-gramas, es mucho menor tanto en tiempo como en espacio respecto a otros algoritmos de minado de secuencias. Y por otro lado en este experimento se utiliza un conjunto de Whiteware a manera de estrategia de detección de malware, y debido a que la sensibilidad del Whiteware es del 100% sin importar el algoritmo esto nos permite decir que el enfoque propuesto detecta la presencia de malware.

Finalmente, como trabajo futuro creemos que el rendimiento del enfoque actual se puede mejorar aumentando el número de muestras y de fuentes de información por cada tipo de malware; y de esta manera la clasificación sería más precisa y no sólo se contemplaría el tipo de malware sino también la familia a la que pertenece. Por otro

lado se puede considerar otros experimentos de clases no balanceadas de malware, es decir con variantes de malware con una similitud vectorial menor.

Referencias

1. Shijo, P.V., Salim, A.: Integrated Static and Dynamic Analysis for Malware Detection. Procedia Comput. Sci., Vol. 46, No. Icict, pp. 804–811 (2015)
2. Moser, A., Kruegel, C., Kirda, E.: Limits of Static Analysis for Malware Detections.
3. Zhao, H., Xu, M., Zheng, N., Yao, J., Hou, Q.: Malicious executables classification based on behavioral factor analysis. In: IC4E, International Conference on e-Education, e-Business, e-Management and e-Learning, pp. 502–506 (2010)
4. Ahmadi, M., Sami, A., Rahimi, H., Yadegari, B.: Malware detection by behavioural sequential patterns. Comput. Fraud Secur., Vol. 2013, No. 8, pp. 11–19 (2013)
5. Shcherbina, V.S., Zakharov, V.A.: Using algebraic models of programs for detecting Metamorphic Malwares. r . I . Podlovchenko, n . N . Kuzyurin. Vol. 172, No. 5, pp. 740–751 (2011)
6. Wang, C., Pang, J., Zhao, R., Fu, W., Liu, X.: Malware detection based on suspicious behavior identification. Proceedings of the 1st International Workshop on Education Technology and Computer Science, ETCS, Vol. 2, pp. 198–202 (2009)
7. Tian, R., Islam, R., Batten, L., Versteeg, S.: Differentiating malware from cleanware using behavioural analysis. In: Proceedings of the 5th IEEE International Conference on Malicious and Unwanted Software, Malware, pp. 23–30 (2010)
8. Egele, M., Scholte, T., Kirda, E., Kruegel, C.: A survey on automated dynamic malware-analysis techniques and tools. ACM Computing Surveys, Vol. 44, No. 2. pp. 1–42 (2012)
9. Islam, R., Tian, R., Batten, L.M., Versteeg, S.: Classification of malware based on integrated static and dynamic features. Journal of Network and Computer Applications, Vol. 36, No. 2. pp. 646–656 (2013)
10. Jonathan, P., Vázquez, B.: PoC : Captura de malware con el honeypot Dionaea - Parte I. Rev. .Seguridad UNAM CERT (2015)
11. Sochor, T., Zuzcak, M.: Study of Internet Threats and Attack Methods Using Honeypots and Honeynets. Computer Networks SE-12, Vol. 431, Kwiecień, A., Gaj, P., Stera, P. Eds. Springer International Publishing, pp. 118–127 (2014)
12. Wa, R., Hunt, G., Brubacher, D.: Detours: Binary Interception of Win32 Functions. Proc. 3rd USENIX Wind. NT Symp., pp. 135–143 (1999)
13. Bayer, U., Kruegel, C., Kirda, E.: TTAnalyze : A Tool for Analyzing Malware.
14. Bayer, U., Moser, A., Kruegel, C., Kirda, E.: Dynamic analysis of malicious code. pp. 67–77 (2006)
15. Willems, G., Holz, T., Freiling, F.: Toward automated dynamic malware analysis using CWSandbox. IEEE Secur. Priv., Vol. 5, No. 2, pp. 32–39 (2007)
16. Guarnieri, C.: CuckooSandbox. [Online]. Available: http://www.cuckoosandbox.org/about.html. [Accessed: 01-Jun-2015].
17. Gheorghescu, M.: An Automated Virus Classification System. Virus Bull. Conf., no. October, pp. 294–300 (2005)
18. Rieck, K., Trinius, P., Willems, C., Holz, T.: Automatic Analysis of Malware Behavior using Machine Learning. pp. 1–30 (2011)
19. Wysopal, C., Shields, T.: Static Detection of Application Backdoors Detecting both malicious software behavior and malicious, Info.
20. Syntactic Clustering of the Web. [Online]. Available: http://www.hpl.hp.com/techreports/Compaq-DEC/SRC-TN-1997-015.pdf. [Accessed: 15-Dec-2015].

21. Sidorov, G.: Syntactic Dependency Based N-grams in Rule Based Automatic English as Second Language Grammar Correction. Int. J. Comput. Linguist. Appl., Vol. 4, No. 2, pp. 169–188 (2013)
22. Windows API Index (Windows). [Online]. Available: https://msdn.microsoft.com/en-us/library/windows/desktop/ff818516(v=vs.85).aspx. [Accessed: 16-Dec-2015]
23. Support-Vector Networks. [Online]. Available: http://image.diku.dk/imagecanon/material/cortes_vapnik95.pdf. [Accessed: 15-Dec-2015].
24. Cuckoo Sandbox: Malware repository. [Online]. Available: https://malwr.com/. [Accessed: 02-Jun-2015]
25. Bontchev, V., Software, F., Thverholt, I.: Current Status of the CARO Malware Naming Scheme.
26. Narouei, M., Ahmadi, M., Giacinto, G., Takabi, H., Sami, A.: DLLMiner : structural mining for malware detection. No. April, pp. 3311–3322 (2015)

Compilación de un lexicón de redes sociales para la identificación de perfiles de autor

Helena Gómez-Adorno, Ilia Markov, Grigori Sidorov,
Juan-Pablo Posadas-Duran, Carolina Fócil-Arias

Instituto Politécnico Nacional,
Centro de Investigación en Computación,
México

helena.adorno@gmail.com, markovilya@yahoo.com, sidorov@cic.ipn.mx,
jpposadas@gmail.com, focil.carolina@gmail.com

Resumen. En este trabajo presentamos un recurso léxico para el preprocesamiento de textos publicados en redes sociales desarrollado para los idiomas: inglés, español, holandés e italiano. El recurso se compone de diccionarios de palabras *slang*, abreviaturas, contracciones y emoticones utilizados comúnmente en redes sociales. Los diccionarios fueron utilizados en el preprocesamiento de *tweets* obtenidos del corpus de la competencia de identificación de perfiles de autor del PAN 2015 y los resultados demuestran que el uso de los diccionarios ayuda a mejorar la eficiencia de los clasificadores para la tarea de identificación de perfiles de autor.

Palabras clave: Lexicón, redes sociales, perfil de autor, clasificación de textos.

Compiling a Lexicon of Social Media for the Author Profiling task

Abstract. In this paper, we present a lexical resource for preprocessing of texts published in social networks. It is developed for the following languages: English, Spanish, Dutch, and Italian. The resource contains dictionaries of slang words, abbreviations, contractions, and emoticons commonly used in social networks. The dictionaries were used for preprocessing of tweets obtained from the corpus for the task of author profiling (PAN 2015). The results show that the use of the proposed dictionaries helps to improve the efficiency of classifiers for the author profiling task.

Keywords: Lexicon, social networks, author profiling, text classification.

1. Introducción

El uso de las redes sociales está en incremento constante a nivel mundial. Cientos de usuarios se inscriben a diario en las diferentes plataformas existentes, por lo tanto, el contenido extraído de las redes sociales es fundamental para tareas como análisis de sentimiento [11], detección de perfiles de autores [13], identificación de autores [9,14], minería de opiniones [4], detección de plagio [17], cálculo de similitud entre textos [18,20] y para desarrollar sistemas robustos que ayuden a la toma de decisiones en áreas relacionadas como la política, la educación, la economía, entre otras.

El procesamiento de los mensajes publicados en redes sociales no es una tarea sencilla de resolver [12,2]. Los mensajes publicados en éstas plataformas son generalmente cortos (cientos de palabras) y no siguen las reglas convencionales del idioma, por ejemplo, para componer los textos se utilizan con frecuencia palabras *slang*, abreviaturas y emoticones [8]. Las palabras consideradas como *slang* y las abreviaturas son específicas para cada idioma, y por lo tanto, los sistemas que realizan procesos sobre mensajes de redes sociales necesitan diccionarios específicos.

El objetivo de este trabajo es compilar diccionarios de palabras *slang*, abreviaturas, contracciones y emoticones para ayudar al preprocesamiento de textos publicados en redes sociales. Con el uso de estos diccionarios se pretende mejorar los resultados de las tareas relacionadas con datos obtenidos de dichas plataformas. Por lo tanto, evaluamos nuestra hipótesis en la tarea de identificación de perfiles de autor *(author profiling)*. El objetivo de esta tarea es obtener información respecto al autor de un texto, específicamente su edad y género, analizando mensajes publicados por el autor en Twitter [13,16].

Este trabajo está dividido de la siguiente forma: La sección 2 presenta los detalles de las investigaciones realizadas en el área Procesamiento de Lenguaje Natural y preprocesamiento de textos. La sección 3 describe el procedimiento para la compilación de los diccionarios y la estructura de los mismos. La sección 4 presenta la evaluación de la tarea de identificación de perfiles de autor utilizando los diccionarios desarrollados. Las conclusiones y el trabajo a futuro son presentados en la sección 5.

2. Trabajo relacionado

En esta sección, se presentan algunos de los principales trabajos que demuestran la importancia de la fase de preprocesamiento de datos en diferentes tareas de procesamiento automático de textos. Un correcto preprocesamiento conlleva a un análisis adecuado y ayuda a incrementar la precisión y la eficiencia de los procesos de análisis de textos. Algunos de los retos encontrados a la hora de realizar preprocesamiento de textos de redes sociales son presentados a detalle en el trabajo de Baldwin [3].

En el estudio desarrollado por Clark y Araki [7] se discuten los problemas relacionados con el procesamiento de mensajes obtenidos a partir de redes

sociales. Los autores mejoraron el rendimiento de un corrector ortográfico de código abierto sobre datos de Twitter, mediante el desarrollo de un sistema de preprocesamiento automático para la normalización de dichos datos. Los resultados reportados indican que el sistema es capaz de disminuir el promedio de error por mensaje de 15 % a 5 %.

El trabajo realizado por Hemalatha *et al.* [11] presenta algunos de los pasos de preprocesamiento de textos que deben ser tomados en cuenta para mejorar la calidad de los mensajes obtenidos a través de Twitter. Entre las técnicas mencionadas se encuentran: remover URLs, caracteres especiales, letras repetidas de una palabra y palabras de preguntas (qué, cuándo, como, etc.). Este estudio demostró que al realizar los pasos mencionados anteriormente, el resultado de la tarea de análisis de sentimiento mejora considerablemente.

En la investigación realizada por Haddi *et al.* [10] se utilizó una combinación de diferentes técnicas de preprocesamiento tales como limpieza de etiquetas HTML, expansión de abreviaturas, manejo de palabras de negación, eliminación de palabras auxiliares (*stop words*) y uso de métodos para reducir una palabra a su raíz. El objetivo de este trabajo es el de analizar los sentimientos sobre opiniones relacionadas con películas. Los autores reportaron que un preprocesamiento de textos apropiado puede mejorar el desempeño del clasificador y aumentar los resultados considerablemente en la tarea de análisis de sentimientos.

En [15] se propone la corrección ortográfica de los mensajes encontrados en redes sociales. Esto incluye letras repetidas, vocales omitidas, sustitución de letras con números (típicamente sílabas), uso de ortografía fonética, uso de abreviaturas y siglas. En un enfoque dirigido por datos *data-driven approach* [5] se aplica un filtro de URL combinándolo con técnicas estándar de preprocesamiento de textos.

Como puede observarse, existen diversas investigaciones relacionadas con el preprocesamiento de textos publicados en redes sociales. En este trabajo, se presenta un recurso léxico y se demuestra su importancia para la tarea de identificación de perfiles de autor. En la siguiente sección se describe el procedimiento utilizado para la compilación de los diccionarios y se muestran ejemplos del contenido de los mismos.

3. Creación del lexicón de redes sociales

Este trabajo de investigación comprende el análisis y la recopilación de vocabulario abreviado (utilizado en redes sociales) para la creación de diccionarios en varios idiomas como el inglés, español, holandés e italiano. Los diccionarios fueron recopilados para estos cuatro idiomas ya que son necesarios para el preprocesamiento de *tweets* para la tarea de identificación de perfil del autor del PAN 2015 [16]. El PAN es un laboratorio de evaluación sobre descubrimiento de plagio, autoría y uso indebido de software social, que se celebra en el marco de la conferencia CLEF[1].

[1] *Conference and Labs of the Evaluation Forum:* `http://www.clef-initiative.eu/`

El tipo de vocabulario acortado que generalmente se utiliza en las redes sociales se pueden dividir en tres categorías: palabras *slang*, abreviaturas y contracciones. A continuación se describe brevemente cada categoría:

Palabras *Slang* vocabulario estructurado en una lengua dada, que generalmente se utiliza entre personas del mismo grupo social. Es un metalenguaje que se usa para enriquecer las expresiones, y las palabras tienen una representación fonológica intacta. Algunos ejemplos de palabras *slang* encontrados en el idioma español son bb (bebé), xq (porque), dnd (dónde), tb (también), tqm (te quiero mucho) y xfa (por favor).

Abreviaciones son representaciones ortográficas de una palabra o frase. También se incluyen en esta categoría los acrónimos, los cuales se forman a partir de las letras iniciales de un nombre o partes de palabras o frases. Dentro de esta categoría podemos encontrar los siguientes ejemplos: Arq. (Arquitecto), Sr. (Señor), NY (Nueva York), kg. (kilogramo), Av. (Avenida), entre otras.

Contracciones ocurren cuando dos palabras se reducen en una sola y un apóstrofe toma el lugar de la letra que falta. Hay muchas reglas entre las lenguas para crear contracciones. Sin embargo, esta investigación no tendrá en cuenta ninguna de ellas. Ejemplos de contracciones son: al (a el) y del (de el).

Otro tipo de elemento que aparece con frecuencia en los mensajes de redes sociales son los emoticones. Los emoticones son visualizaciones tipográficas que permiten representar las expresiones faciales de las emociones, es decir, es una manera de darle una carga emotiva a un texto. Se incluyeron dos estilos de emoticones conocidos como occidental y oriental. El estilo occidental se utiliza comúnmente en los Estados Unidos y Europa, los emoticones de este estilo se escriben de izquierda a derecha, como si una cara se gira 90 grados hacia la derecha. Los emoticones mostrados a continuación pertenecen a este estilo: :-) (cara sonriente), :-/ (cara dudosa) y :-o (cara sorprendida). Por el otro lado, se tienen a los emoticones de tipo oriental que son populares en el este de Asia y a diferencia del estilo occidental, los emoticones orientales no se rotan. En este estilo, los ojos son a menudo vistos como una característica importante de la expresión. Algunos ejemplos de este estilo son (^v^) (cara sonriente), ((+_+)) (cara dudosa) y (o.o) cara sorprendida.

En este trabajo realizamos la recopilación de vocabulario abreviado y emoticones que se utilizan generalmente en redes sociales. A continuación se describe el proceso de compilación de los diccionarios:

1. Búsqueda e identificación de sitios web que se utilizan como fuente para la extracción de las listas de palabras *slang*, abreviaturas y contracciones en los cuatro idiomas (inglés, español, italiano y holandés).
2. Extracción manual o semi-automática de todas las palabras *slang*, abreviaturas y contracciones junto con sus respectivos significados de cada sitio web en los diferentes idiomas.
3. Identificación y fusión de todos los archivos de la misma categoría. Limpieza, formateo y estandarización de cada archivo, eliminando duplicados. Verificación manual de significados de cada entrada de los diccionarios.

Mediante el proceso descrito anteriormente se crearon doce diccionarios, divididos en cuatro idiomas, uno para cada categoría (palabras *slang*, abreviaturas y contracciones). Los diccionarios están disponibles de manera gratuita en nuestro sitio web[2], donde además se presenta una breve descripción de los diccionarios, una lista de sitios web utilizados para la recolección de las tres categorías de vocabulario para los cuatro idiomas, y la lista sitios web usados para obtener los emoticones. En el caso del diccionario de palabras *slang* en español también se incluyeron entradas del trabajo [6], en el que se realizó una extracción manual de palabras *slang* de una colección de mensajes de Twitter.

Cada diccionario se ha almacenado en un archivo diferente, los elementos se encuentran ordenados de manera alfabética y la información se codifica usando dos columnas separadas por una tabulación. La primera columna corresponde a una entrada de palabra *slang*, abreviatura o contracción, según sea la naturaleza del diccionario, y la segunda columna corresponde al significado de la entrada correspondiente.

La Tabla 1 presenta las estadísticas de cada diccionario, donde se puede observar que existe un número significativo de palabras *slang* disponibles para inglés y español, mientras que para el caso del holandés e italiano el número de entradas es menor. Por otro lado, se puede observar que hay un gran número de abreviaturas en el idioma holandés. El número total de entradas en nuestro lexicón de redes sociales es de 7,212.

Tabla 1. Número de entradas en cada diccionario

Tipo de diccionario	Holandés	Italiano	Inglés	Español
Abreviaturas	1,237	107	1,346	527
Slangs	250	362	1,249	939
Contracciones	15	56	131	11
Emoticones	-	-	482	482
Totales	1,520	525	3,208	1,959

4. Caso de estudio: Identificación de perfiles de autor

La tarea de identificación de perfiles de autor consiste en la identificación de algunos aspectos de una persona como su edad, sexo, o algunos rasgos de comportamiento basados en el análisis de muestras de texto. El perfil de un autor puede ser utilizado en muchas áreas, por ejemplo, en las ciencias forenses para obtener la descripción de un sospechoso mediante el análisis de los mensajes publicados en redes sociales, y en las empresas para personalizar los anuncios que aparecen en las redes sociales o enviados por medio de correo electrónico[1].

[2] `http://www.cic.ipn.mx/~sidorov`

En los últimos años, se han propuesto diferentes métodos para abordar la tarea de identificación de perfiles de autor, la mayoría de ellos utilizan técnicas de aprendizaje automático, minería de datos y procesamiento del lenguaje natural. Desde un punto de vista de aprendizaje automático, la tarea identificación de perfiles de autor puede ser considerada como un problema de clasificación multi-clase y multi-etiqueta, donde cada elemento S_i de un conjunto de muestras de texto $\mathbf{S} = \{S_1, S_2, \ldots, S_i\}$ se le asignan múltiples etiquetas $(l_1, l_2, \ldots, l_k)$, cada una de ellas representando un aspecto del autor (género, edad, rasgos de comportamiento) y el valor asignado en cada etiqueta representa una categoría dentro del aspecto correspondiente. El problema se traduce a la construcción de un clasificador M que asigna varias etiquetas a los textos no etiquetados.

El enfoque basado en aprendizaje automático está dividido en dos etapas: entrenamiento y prueba. En la etapa de entrenamiento, se obtiene una representación vectorial de cada uno de los textos de ejemplo de cada categoría, es decir, $v^i = \{v_1, v_2, \ldots, v_j\}$ donde v^i es la representación vectorial del texto de ejemplo S_i.

Luego, un clasificador es entrenado utilizado la representación vectorial de las muestras etiquetadas. En este trabajo utilizamos un clasificador basado en máquinas de soporte vectorial (SVM) y generamos diferentes modelos de clasificación para cada uno de los aspectos del perfil de un autor, es decir, aprendemos un modelo para determinar la edad y otro modelo para determinar el género de un autor.

Las características utilizadas en este trabajo se basan en una representación vectorial de la frecuencia de ocurrencia de palabras usando el modelo estándar de bolsa de palabras (en inglés, *Bag of words (BOW)*), que ha demostrado ser efectivo en tareas relacionadas con la caracterización de autores en trabajos previos [19]. En este artículo se utilizan solamente la frecuencia de palabras que ocurren en el conjunto de textos de entrenamiento para construir el modelo de representación.

En la fase de prueba o evaluación, la representación vectorial de los textos no etiquetados es obtenida utilizando las mismas características extraídas en la etapa de entrenamiento. Luego, se utiliza el clasificador para asignar valores a las etiquetas de cada aspecto del perfil del autor de cada usuario del conjunto de prueba.

Con el objeto de evaluar la utilidad de nuestros diccionarios, utilizamos el corpus diseñado para la tarea identificación de perfiles de autor del PAN 2015. El corpus está compuesto de *tweets* en cuatro idiomas diferentes: inglés, español, italiano y holandés. Cada idioma tiene un conjunto de *tweets* etiquetados que corresponden a la edad y género del autor de dicho *tweet*. Los valores de las etiquetas de la clase género pueden ser: hombre o mujer. Los valores de las etiquetas de la clase edad pueden ser: 18-24, 25-34, 35-49, 50-xx.

El corpus de identificación de perfiles de autor del PAN-2015 está parcialmente disponible. Debido a la política de los organizadores, sólo el corpus de entrenamiento ha sido liberado. En este sentido, se realizaron los experimentos

Tabla 2. Resultados obtenidos para la clasificación de género

Language	**SVM Liblinear**	
	sin preprocesamiento	**con preprocesamiento**
Inglés	74.91	**76.33**
Español	80.00	**81.00**

Tabla 3. Resultados obtenidos para la clasificación de edad

Language	**SVM Liblinear**	
	sin preprocesamiento	**con preprocesamiento**
Inglés	75.14	**76.31**
Español	68.70	**69.11**

utilizando el corpus de entrenamiento y se realizó validación cruzada de 10 capas para evaluar nuestra propuesta.

Las tablas 2 y 3 presentan la exactitud obtenida para las clases género y edad respectivamente, con y sin preprocesamiento del corpus. Podemos concluir que para cada lenguaje, los mejores resultados fueron obtenidos cuando se realiza el preprocesamiento utilizando nuestros diccionarios.

La etapa de preprocesamiento consiste básicamente en la identificación dentro del corpus de palabras que se encuentren en nuestros diccionarios y reemplazarlas por sus respectivos significados. Cabe mencionar que para este trabajo no realizamos ningún proceso de desambiguación del sentido de las palabras y por tanto, solo se selecciona el primer significado disponible para cada término.

5. Conclusiones y trabajo futuro

En este trabajo presentamos un lexicón de redes sociales que contiene diccionarios de palabras *slang*, abreviaturas, contracciones y emoticones más populares en las redes sociales. El recurso contiene diccionarios en idioma inglés, español, holandés, e italiano. Además, describimos la metodología de la recopilación de datos, listamos las direcciones URL utilizadas como fuentes para la creación de cada diccionario, y explicamos el proceso de estandarización de los mismos. Luego, proporcionamos información relativa a la estructura de los diccionarios y una descripción de la longitud de cada uno de ellos.

Al momento de utilizar los diccionarios para preprocesamiento de textos nos dimos cuenta de que hay algunos términos que se usan comúnmente en las redes sociales que no están presentes en nuestras fuentes web, especialmente para los idiomas inglés, italiano y holandés. Por lo tanto, para un trabajo futuro, tenemos la intención de ampliar los diccionarios de palabras *slang* con entradas

recogidas manualmente para cada idioma, de la misma manera que se hizo para el diccionario de palabras *slang* en español.

Agradecimientos. Este trabajo ha sido realizado gracias al apoyo de la "Red Temática en Tecnologías del Lenguaje - CONACYT" y Gobierno Mexicano (Proyecto CONACYT 240844, SNI, COFAA-IPN, SIP-IPN 20151406, 20161947).

Referencias

1. Argamon, S., Koppel, M., Pennebaker, J.W., Schler, J.: Automatically profiling the author of an anonymous text. Commun. ACM 52(2), 119–123 (2009)
2. Atkinson, J., Figueroa, A., Pérez, C.: A semantically-based lattice approach for assessing patterns in text mining tasks. Computación y Sistemas 17(4), 467–476 (2013)
3. Baldwin, T.: Social media: Friend or foe of natural language processing? In: 26th Pacific Asia Conference on Language, Information and Computation. pp. 58–59 (2012)
4. Ben-Ami, Z., Feldman, R., Rosenfeld, B.: Using Multi-View Learning to Improve Detection of Investor Sentiments on Twitter. Computación y Sistemas 18, 477–490 (2014)
5. Brigadir, I., Greene, D., Cunningham, P.: Adaptive Representations for Tracking Breaking News on Twitter. ArXiv e-prints (2014)
6. Camacho-Vázquez, V., Sidorov, G., Galicia-Haro, S.N.: Machine learning applied to a balanced and emotional corpus of tweets with many varieties of Spanish. submitted (2016)
7. Clark, E., Araki, K.: Text normalization in social media: Progress, problems and applications for a pre-processing system of casual English. Procedia - Social and Behavioral Sciences 27, 2–11 (2011)
8. Das, D., Bandyopadhyay, S.: Document Level Emotion Tagging: Machine Learning and Resource Based Approach. Computación y Sistemas 15, 221–234 (2011)
9. Gómez-Adorno, H., Sidorov, G., Pinto, D., Markov, I.: A graph based authorship identification approach: Notebook for PAN at CLEF 2015. In: Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, Toulouse, France, September 8-11, 2015. (2015)
10. Haddi, E., Liu, X., Shi, Y.: The role of text pre-processing in sentiment analysis. Procedia Computer Science 17, 26–32 (2013), first International Conference on Information Technology and Quantitative Management
11. Hemalatha, I., Varma, D.G.P.S., Govardhan, D.A.: Preprocessing the informal text for efficient sentiment analysis. International Journal of Emerging Trends & Technology in Computer Science (IJETTCS) 1(2), 58–61 (2012)
12. Pinto, D., Vilariño-Ayala, D., Alemán, Y., Gómez-Adorno, H., Loya, N., Jiménez-Salazar, H.: The soundex phonetic algorithm revisited for sms-based information retrieval. In: II Spanish Conference on Information Retrieval CERI 2012 (2012)
13. Posadas-Durán, J.P., Gómez-Adorno, H., Markov, I., Sidorov, G., Batyrshin, I.Z., Gelbukh, A.F., Pichardo-Lagunas, O.: Syntactic n-grams as features for the author profiling task: Notebook for PAN at CLEF 2015. In: Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, Toulouse, France, September 8-11, 2015. (2015), `http://ceur-ws.org/Vol-1391/136-CR.pdf`

14. Posadas-Duran, J.P., Sidorov, G., Batyrshin, I.: Complete syntactic n-grams as style markers for authorship attribution. In: Human-Inspired Computing and Its Applications, pp. 9–17. Springer (2014)
15. Rangarajan Sridhar, V.K.: Unsupervised text normalization using distributed representations of words and phrases. In: Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing. pp. 8–16. Association for Computational Linguistics, Denver, Colorado (2015), http://www.aclweb.org/anthology/W15-1502
16. Rangel, F., Rosso, P., Potthast, M., Stein, B., Daelemans, W.: Overview of the 3rd author profiling task at pan 2015. CLEF (2015)
17. Sanchez-Perez, M.A., Gelbukh, A., Sidorov, G.: Adaptive algorithm for plagiarism detection: The best-performing approach at pan 2014 text alignment competition. In: Mothe, J., Savoy, J., Kamps, J., Pinel-Sauvagnat, K., Jones, J.G., SanJuan, E., Cappellato, L., Ferro, N. (eds.) Experimental IR Meets Multilinguality, Multimodality, and Interaction: 6th International Conference of the CLEF Association, CLEF'15, Toulouse, France, September 8-11, 2015, Proceedings, pp. 402–413. Springer International Publishing, Cham (2015)
18. Sidorov, G., Gómez-Adorno, H., Markov, I., Pinto, D., Loya, N.: Computing text similarity using tree edit distance. In: Fuzzy Information Processing Society (NAFIPS) held jointly with 2015 5th World Conference on Soft Computing (WConSC), 2015 Annual Conference of the North American, Redmond, WA, USA. pp. 1–4 (2015)
19. Stamatatos, E.: A survey of modern authorship attribution methods. Journal of the American Society for Information Science and Technology 60(3), 538–556 (2009)
20. Vilariño, D., Pinto, D., León, S., Alemán, Y., Gómez-Adorno, H.: Buap: N-gram based feature evaluation for the cross-lingual textual entailment task. Atlanta, Georgia, USA p. 124 (2013)

Sistema de traducción directa de español a LSM con reglas marcadas

Obdulia Pichardo-Lagunas, Luis Partida-Terrón, Bella Martínez-Seis, Adriana Alvear-Gallegos, Raúl Serrano-Olea

Instituto Politécnico Nacional,
Unidad Profesional Interdisciplinaria en Ingeniería y Tecnologías Avanzadas,
Ciudad de México, México

{opichardol, bmartinezs}@ipn.mx, lpartidat1000@alumno.ipn.mx

Resumen. La traducción automática se enfrenta a complejidades computacionales y lingüísticas; así como a los principios que rigen tanto la lengua origen como la lengua meta. Este proceso se complica aún más alguna de las lenguas involucradas no son escritas como es el caso de la Lengua de Señas Mexicana (LSM), la cual es desarrollada por personas con discapacidad auditiva. El presente trabajo se centra en el desarrollo de una herramienta para la traducción directa con reglas marcadas del español escrito a Lengua de Señas Mexicano (LSM). Para ello se hace uso de bases de datos multimedia por ser una lengua viso-gestual y de PNL para traducción automática con análisis léxico, sintáctico y morfológico.

Palabras clave: Traducción automática, sordera, procesamiento de lenguaje natural, lingüística computacional, bases de datos multimedia, LSM.

System for Direct Translation from Spanish into LSM with Marked Rules

Abstract. Automatic translation faces computational as well as linguistic complexities because of the principles and standards of the original and the target language. This process is even bigger challenge when the languages are not written, as it is the case of Mexican Sign Language (LSM), which is used by people with hearing disabilities. This paper focuses on the development of a tool for direct translation with rules; the translation is from written Spanish to Mexican Sign Language. This tool uses multimedia databases because of the visual-gestual language involved, and it uses NLP for the automatic translation with lexical, syntactic and morphological analysis.

Keywords: Automatic translation, deafness, natural language processing, computational linguistics, multimedia databases, LSM.

1. Introducción

La discapacidad auditiva se refiere a la falta o reducción de la habilidad para oír claramente debido a un problema en algún lugar del mecanismo auditivo [1]. El uso de

la lengua de señas sitúa, a las personas sordas, como una comunidad lingüística minoritaria, quienes hacen uso de la lengua de señas mexicana (LSM) como medio de comunicación. Dicha lengua no tiene una relación directa con el español de México por lo que la población oyente no se puede comunicar fácilmente por escrito con la comunidad siliente.

Las personas sordas son clasificadas según su manejo del lenguaje en seis grupos [2]: el primer grupo lo conforman los **monolingües** que solo se expresan en LSM, el segundo los que tienen **como primera lengua la LSM** y como una segunda lengua el español oral y/o escrito, el tercer grupo está formado por los **bilingües** que además de la LSM usan alguna otra lengua de señas como la ASL, el cuarto grupo conocen la LSM y **el español puede considerarse su primera lengua, e**n el quinto grupo habría que considerarse a los **semilingües** que no son competentes en la LSM ni en español y el sexto grupo lo constituyen aquellos que no han recibido educación formal y por lo tanto **desconocer el español y la LSM** usando como medio de comunicación señas caseras o familiares. De tal forma que la mayoría de los sordos en México enfrentan dificultades para entablar una comunicación con una persona en español.

La LSM no cuenta con un vocabulario establecido oficialmente ni con reglas gramaticales normadas, ya que en ocasiones es desarrollada de manera intuitiva por sectores de la población con discapacidad auditiva. Además, la educación de las personas sordas depende del momento en el cual perdieron la capacidad auditiva: las personas que se volvieron sordas después de haber adquirido el español no presentan un problema lingüístico, porque, en el español escrito no son diferentes a los oyentes; en cambio, las personas que perdieron la audición antes de adquirir la lengua de su comunidad, en este caso el español, pocos llegan a tener una competencia lingüística en esa lengua [14].

Las principales diferencias entre las lenguas de señas y las lenguas orales son principalmente en su estructura, ya que es ágrafa (sin escritura) y no se produce (emisión vocal) ni se percibe (atención auditiva) como las lenguas orales. La Lengua de Señas basa su funcionamiento en la percepción visual; posee su propio vocabulario y elementos llamados parámetros formacionales, que son las diferentes partes que forman un signo (por ejemplo, el movimiento de la mano, el lugar donde se hace el signo, la forma de la mano, expresión corporal) y estos, a su vez, formarán las frases signadas [12]. Cabe destacar que existen factores que provocan variaciones de la LSM, ya que no se cuenta con diccionarios de LS, hay baja capacitación y enseñanza de la misma, los carentes puntos de reunión entre personas sordas, al igual que los oyentes se utilizan términos y expresiones que identifican a un grupo personas.

La integración de grupos sociales marginados es de suma importancia, por ello es importante que las personas oyentes conozcan y aprendan la Lengua de Señas Mexicana (LSM), de esta forma se rompe la barrera de comunicación que existe entre una persona sorda y una oyente, y el distanciamiento que se genera incluso con su familia. Por lo tanto se considera pertinente el desarrollo de herramientas de apoyo para comprender y/o aprender LSM.

El sistema desarrollado proporciona funciones de análisis léxico, sintáctico y morfológico. La traducción automática se realiza del español de México escrito a la LSM, de manera que se deberá ingresar una oración en español y ésta se mostrará de forma escrita como es utilizada en el LSM, y al ser una lengua viso-gestual se desplegará el video con las señas de la frase traducida. Para realizar esto debemos

indicar unos temas de interés los cuales se muestran en la sección de Antecedentes, en la Sección 2 está la metodología, aquí se describen las reglas para realizar la traducción además de cuestiones básicas para el funcionamiento del traductor, en la Sección 3 se muestran los resultados obtenidos con algunas frases obtenidas y finalmente en las conclusiones hacemos un análisis de lo logrado y los aspectos a mejorar.

2. Antecedentes

En esta sección se describen los antecedentes y elementos esenciales del procesamiento natural del lenguaje empleados, algunos conceptos de traducción directa y de la gramática usada por LSM.

El Procesamiento de Lenguaje Natural (PLN) es el reconocimiento y utilización de la información expresada en un lenguaje humano a través de sistemas informáticos [6]. Algunos tipos de análisis en PLN incluyen: el léxico, el morfológico, el sintáctico, el semántico, de discurso, entre otros; de los cuales hacemos uso de los tres primeros en este acercamiento a la traducción directa con las reglas marcadas. El análisis léxico especifica los *tokens* del lenguaje considerando que puede haber varios tokens que correspondan a una misma expresión regular. El problema de reconocer *que* una palabra, por ejemplo, en plural, digamos *zapatos* se puede descomponer en morfemas (“zapato” y “–s”) y construir una representación estructurada de tal descomposición, se conoce como análisis morfológico. Un análisis sintáctico o *parsing* es la combinación del reconocimiento de una cadena (oración) de entrada con la asignación a ella de una estructura sintáctica. Se suele representar tal estructura (o derivación) mediante un árbol [6]. Sin embargo también se utiliza para la identificación de los componentes funcionales de las oraciones.

El PLN incluye a los sistemas traductores automáticos. Mediante ellos, un usuario puede leer, en su propio lenguaje, un texto escrito en otro lenguaje; o conversar, de forma escrita u oral, con otros que no comparten su misma lengua [3]. Una clasificación ya clásica de los sistemas de TA establece tres grandes grupos, los enfoques directos, los de interlingua y los de transferencia (normalmente sintáctica y en contados casos también semántica). Dichos grupos conforman la pirámide propuesta por Hutchins & Somers [4] que se basa en las diferencias de "longitudes relativas" de los tres componentes de la traducción: análisis, transferencia y síntesis o generación.

La arquitectura directa o *transformer* se usa cuando no existen teorías lingüísticas formales, que se cuente con lexicones de pocas palabras para la lengua de origen y algunas decenas de reglas gramaticales para dar cuenta de los procesos de desambiguación y de reordenación del texto meta. En su forma más pura, la traducción directa conlleva la traducción palabra por palabra junto con un proceso de equivalencias de cadenas y reordenación del texto meta [5]. Los problemas inherentes a tal metodología son evidentes: el sistema no toma en consideración la estructura sintáctica de la frase ni las relaciones semánticas que existen entre las palabras [7]. Además, no existe ninguna forma de asegurar la correcta formación de las expresiones del lenguaje objeto, ya que no existen reglas gramaticales. Otra característica que limita seriamente las posibilidades de las arquitecturas *transformer* es la total inexistencia de una gramática de la lengua meta.

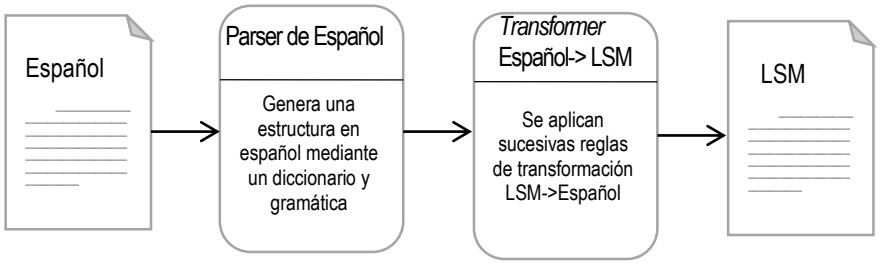

Fig. 1. Traducción Directa de español a LSM.

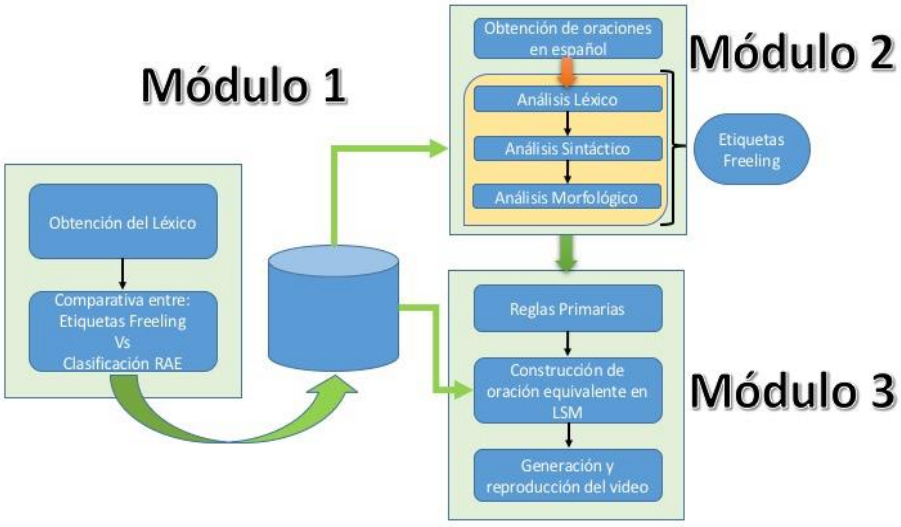

Fig. 2. Arquitectura del sistema por módulos.

La única información que el sistema posee sobre esta lengua son las reglas morfológicas de formación de palabras, además de las reglas de transformación. La Fig. 1 muestra este tipo de proceso con algunas reglas gramaticales añadidas para la traducción español-inglés.

La traducción automática de forma directa es útil en este caso de estudio debido a la falta de normas en LSM. Existen trabajos que han estudiado y propuesto estructuras gramaticales. Andy Eautough [8] quién presenta un panorama general sobre la gramática de la LSM ofreciendo varios aspectos sobre la sintaxis como el sistema pronominal, la negación, oraciones simples y complejas. Boris Fridman es un investigador que se destaca por sus aportaciones al estudio de la comunidad Sorda en México; ha investigado con mayor profundidad los verbos y estructura del lenguaje de la LSM [9]. Ha habido estudios sobre el uso del espacio con valor gramatical Hayawek [10] y la adquisición de la LSM [11]. El vocabulario en LSM tampoco está definido y se enfrenta a la variación geográfica y de lenguas con las que se relaciona, obteniendo una variación léxica.

3. Metodología

El sistema es una herramienta de apoyo para una traducción directa del español a Lengua de Señas Mexicana con determinadas reglas marcadas. Se definió un determinado léxico para recabar de cada palabra la seña que es utilizada en LSM, para este punto se optó por recurrir al apoyo del Centro de Atención Integral a personas con Discapacidad de la Delegación Cuauhtémoc.

El funcionamiento del sistema se muestra en la Fig. 2, donde se aprecian los módulos que lo conforman. El Módulo 1 comprende la parte de generar muestro diccionario de palabras, el Módulo 2 es la parte donde mediante el análisis de una frase en español, se obtienen las etiquetas para su posterior uso, finalmente el Módulo 3 es la sección donde según las reglas básicas que se han determinado, se construye una oración que es equivalente a la forma en que se forman las frases en LSM, de esta nueva oración se genera el video con las señas necesarias.

3.1. Validación de la caracterización dada por la RAE y EAGLE

Este módulo tiene como origen la obtención del léxico, con el cual se realizó una comparación detallada de todo el vocabulario entre la clasificación de la RAE con el etiquetado Eagle.

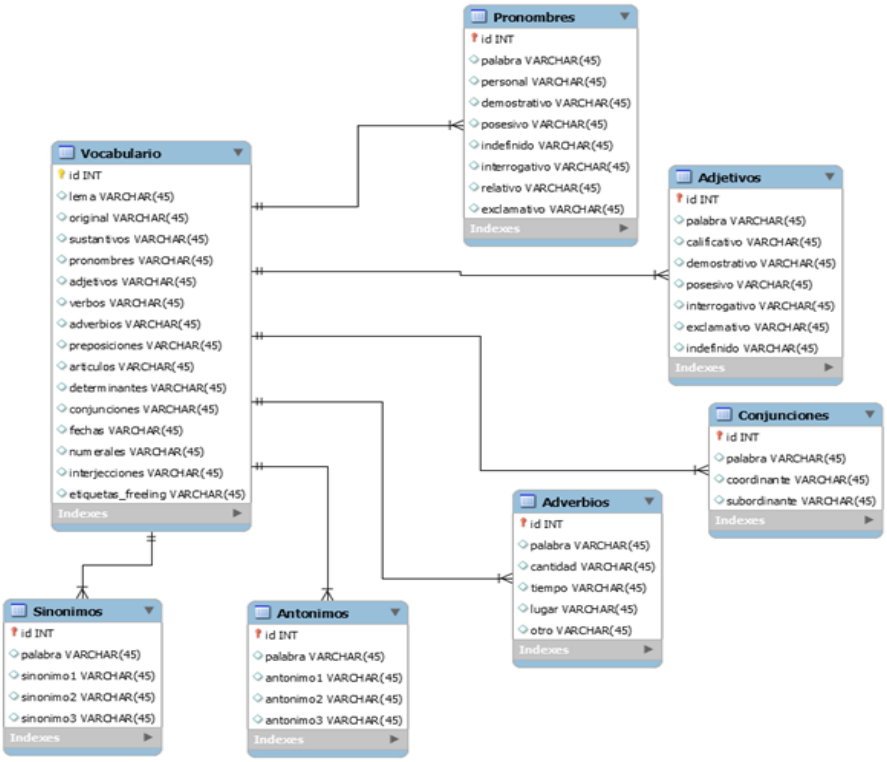

Fig. 3. Diagrama Relacional de la Base de Datos de Etiquetado.

Obtención del léxico: se realizó una investigación del LSM para poder obtener un diccionario con la palabra en español y la grabación en video, de la seña correspondiente.

Comparativa entre etiquetado Freeling y la clasificación RAE: El analizador morfológico para el castellano utiliza un conjunto de etiquetas para representar la información morfológica de las palabras, el cual se basa en las etiquetas propuestas por el grupo EAGLES para la anotación morfosintáctica de oraciones, estas etiquetas no corresponden necesariamente a la clasificación que realiza la Real Academia Española (RAE), para las palabras. Debido a esto fue necesario hacer un etiquetado manual de nuestro diccionario, para obtener su etiqueta EAGLE y su clasificación dada por la RAE, y generar una tabla de equivalencias entre ambas. Estas equivalencias serán ocupadas más adelante para aplicar nuestras reglas de traducción.

Con esto se generó una base de datos (Ver Fig. 3), que se utilizará para buscar las palabras que se deseen traducir, este etiquetado también permite acceder a la tabla comparativa y listado de palabras por etiqueta.

Esta base de datos con el resultado de dicha comparación constituye la base para la realización de la traducción directa con reglas establecidas, las cuales serán descritas a detalle en la siguiente sección.

3.2. Traducción directa

Freeling proporciona funcionalidades de análisis del lenguaje, tales como:

- Tokenización texto,
- Etiquetado de roles,
- División de frases,
- El análisis morfológico
- Tratamiento sufijo, retokenización de los pronombres clíticos,
- Compuesto de reconocimiento de palabras,
- El reconocimiento de múltiples palabras flexibles,
- La predicción probabilística de categorías de palabras desconocidas,
- Codificación fonética,
- Búsqueda basada en SED de palabras similares en el diccionario
- Detección de entidades nombradas
- El reconocimiento de las fechas, números, proporciones, la moneda, y las magnitudes físicas (velocidad, peso, temperatura, densidad, etc.),
- Análisis superficial basado-Chart,
- Clasificación de una denominada sociedad,
- Anotación de sentido y desambiguación basado en WordNet,
- Análisis de dependencias basada en reglas,
- Análisis de dependencias estadística,
- Etiquetado semántico estadístico,
- Resolución de la correferencia,
- Extracción gráfico semántica.

Con el fin de implementar la gramática en forma computacional, se utilizó la herramienta Freeling, que al usarse como una biblioteca externa de nuestra aplicación

nos sirvió principalmente para el análisis léxico (tokenización del texto y las frases ingresadas en la aplicación), análisis sintáctico (para la división de frase se identificación de elementos gramaticales en la oración) y el análisis morfológico de cada una de las palabras. Estos tres análisis son indispensables para el funcionamiento de las reglas a cumplir para la traducción.

La aplicación se desarrolló en JAVA, la cual se encarga de la conexión con el manejador de base de datos creada en MYSQL, donde se almacena la totalidad del vocabulario con sus respectivas imágenes que muestran la señal para cada palabra, la clasificación para las palabras según la RAE, el etiquetado de EAGLE para las mismas y la tabla de equivalencias.

Al no existir una gramática formal en LSM, se diseñó una forma gramatical básica para implementar en este sistema, como se muestra en la Fig. 4**,** que se convertirán en las reglas de traducción.

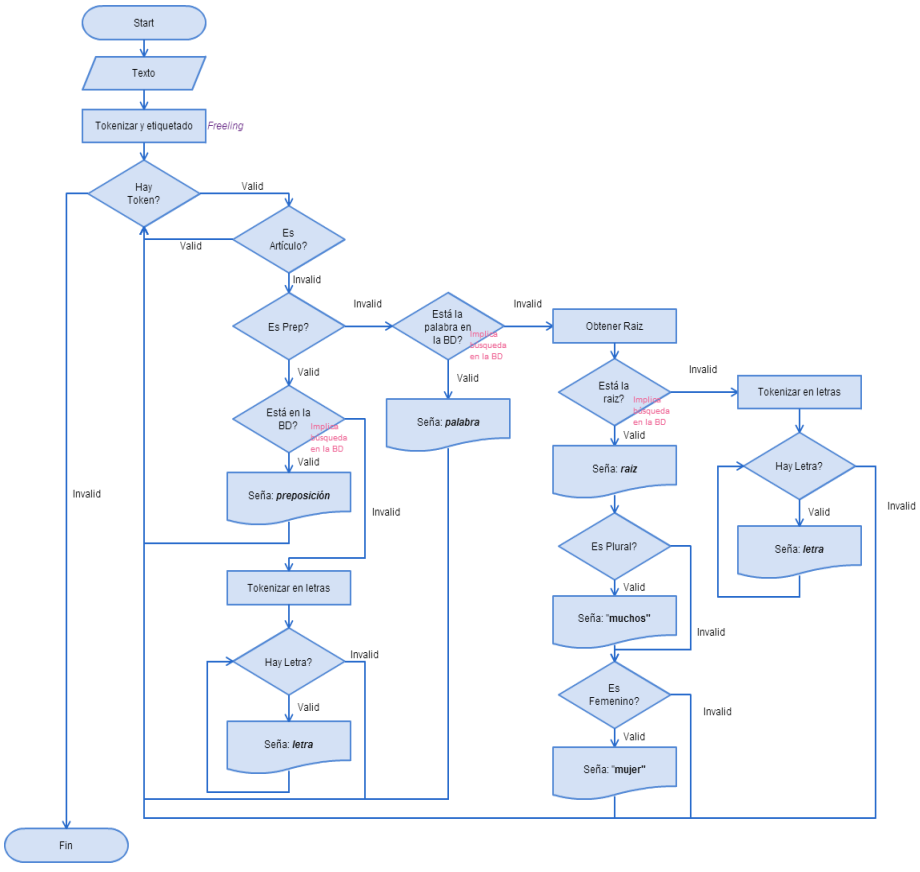

Fig. 4. Representación de la gramática básica a seguir para la traducción.

Como se aprecia en la Fig. 4**.** Se inicia con la captura del texto, una vez ingresada la frase a traducir, se hace la tokenización y el etiquetado de las palabras, el analizador se encarga de clasificar según su sintaxis y morfología. En el análisis sintáctico se toma en cuenta para verificar el cumplimiento de las siguientes reglas:

1) Validación de la frase en idioma español.
2) Tokenización y etiquetado del texto.
3) División de frases para su manejo individual.
4) Los artículos se descartan debido a que en LSM no se utilizan señas para esto.
5) Las preposiciones se buscan en nuestra base de datos, si aparece esta señal será almacenada temporalmente hasta que se termine de realizar la traducción, si no se encuentra localizada en la base de datos se continúa con el proceso de deletreo de la palabra que se explicará más adelante.
6) El resto de las palabras se buscan, si se encuentra, la palabra se agrega a las lista de reproducción del video en LSM.
7) Se obtiene la raíz (lema) de la palabra, si la palabra no está en nuestro vocabulario, y se busca en determinado listado. Cabe destacar que se le agregan un par de vocablos que resultan importantes si son requeridas.
8) A las palabras en plural se le agrega la seña de "muchos".
9) A las palabras en femenino se les agrega la seña de "mujer".
10) Las palabras o lemas que no cumplan con ninguna de las reglas anteriores se deletrean.

En el análisis léxico se verifica la existencia de la palabra (regla 3, 4 y 7) o en su defecto se deletrea (Regla 10). El análisis sintáctico es empleado por si ingresó más de una frase, estas son divididas para su manejo individual, realizado en la regla 3, además se toma en cuenta para verificar el cumplimiento de las reglas 4, 5, 6 y 7. Mientras el análisis morfológico se emplea en la regla 8, 9 y 10 para la obtención del lema y elementos de género y número. Por ejemplo, para decir, "niña", se hace la seña de "niño" más la seña de "mujer". Y en el caso de querer decir, "niñas", se referencia a las señas de "niños" + "muchos" + "mujer". Cabe mencionar que en la regla 9 y 10 se válida que palabras como madre o mamá, no se les agregue por obviedad el término de femenino, ya que, en estos casos solo se requiere de una seña para interpretar la palabra.

Por último si la palabra no cayó en ninguno de los casos anteriores no significa necesariamente que la palabra no exista en LSM, solo que debido a nuestro léxico ajustado para realizar una primera traducción en el ámbito del hogar no ha sido posible agregar un mayor número de palabras, pero si existe la posibilidad real de que la palabra verdaderamente no exista, ya que, el número de palabras existentes en esta lengua es mucho menor a las que se tienen registradas en el español. Por lo cual se procede a deletrear la raíz obtenida en el paso anterior, esta ingresa a un ciclo para separarla carácter por carácter, en donde se agregaran de forma individual a nuestra traducción final, las letras que la conformen. El deletreo en LSM es en ocasiones muy recurrido, por ejemplo, al mencionar nombres propios, marcas, palabras nuevas, o para darse a entender con personas sordas de otras regiones, donde pueden variar las señas de las palabras para un mismo objeto.

El proceso se repetirá con cada palabra que ha sido ingresada y etiquetada, si solo se ingresó una frase al terminar, se tendrá de forma escrita el resultado de la traducción. Si se ingresó más de una frase el proceso continuará hasta terminar la totalidad de las frases. Obteniendo así una traducción por cada frase.

Como se mencionó la lengua de señas basa su funcionamiento en la percepción visual, por lo que no sería un buen traductor, si no tuviera una parte visual donde se aprecie la interpretación de las señas. Por tal motivo con las traducciones obtenidas se obtienen las secuencias de las imágenes que contienen las señas, los archivos se unen

correspondiendo a la frase traducida, y se crea un video temporal que es mostrado con la posibilidad de aumentar o disminuir la velocidad del mismo.

3.3 Reproducción de texto traducido en Lengua de Señas

El sistema proporcionará a cualquier usuario que tenga los conocimientos del español escrito la posibilidad de traducir frases en este idioma a lengua de señas, de esta forma puede servir como un primer acercamiento a esta lengua o para ayudar a entender la forma de aprender de las personas que se comunican mediante LSM.

Por lo cual se diseñó una interfaz de usuario, como se muestra en la Fig. 5. Las imágenes pueden variar con respecto a la última versión de la aplicación.

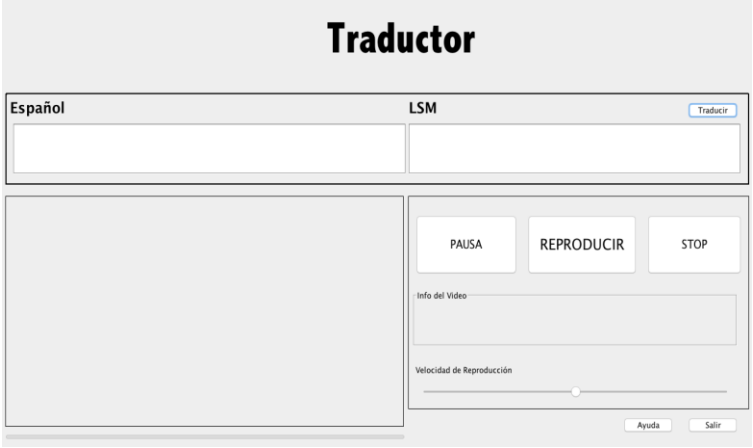

Fig. 5. Interfaz gráfica del sistema.

Como se puede apreciar la interfaz es muy intuitiva, cuenta con un área para escribir la o las frase(s) en español, mientras el área para la parte de la traducción no podrá ser modificada, además cuenta con un botón que se deberá seleccionar para comenzar la traducción. El área del rectángulo de mayor tamaño está destinada para la reproducción del video, y se podrá tener acceso a algunos controles del lado derecho de este. Como son los botones de REPRODUCIR, PAUSA y DETENER, además se cuenta con una barra para aumentar o disminuir la velocidad del video. En la sección de información del video se mostrará el número de videos (dependiendo las frases ingresadas), y cual se está reproduciendo.

Como se menciona su funcionamiento es muy intuitivo, solo se ingresa una frase a traducir en la sección de español y se presiona el botón de traducir. El programa mostrará un mensaje de traducción finalizada al concluir.

4 Resultados

En esta sección se muestran los resultados obtenidos en la durante la ejecución del sistema para la traducción de frases.

Tabla 1. Ejemplos de traducción.

Frase en español	Traducción LSM	Comentario
El niño	Niño	Eliminación del artículo.
Los niños	Niño muchos	Se le agrega la seña de muchos indicando el plural.
La niña	Niño mujer	Se ocupa la palabra raíz que se tiene en la base de datos y se agrega la seña de mujer, indicando el femenino de la palabra.
Mi amigo Juan	Mi amigo J U A N	La palabra que se encuentra separada por espacios (JUAN), se deletreará en LSM, ya que, no se encontró la seña para expresar la palabra, esto sucede con los nombres propios.
Mi casa es grande	Mi casa estar grande	En LSM no existen conjugaciones por lo cual se busca en la base de datos la raíz y esta se muestra en la interfaz.

En la tabla 1 se pueden apreciar ejemplos básicos de los tipos de reglas que se siguen para la traducción. A continuación se mostrarán ejemplos más complejos donde se mezclas estos ejemplos

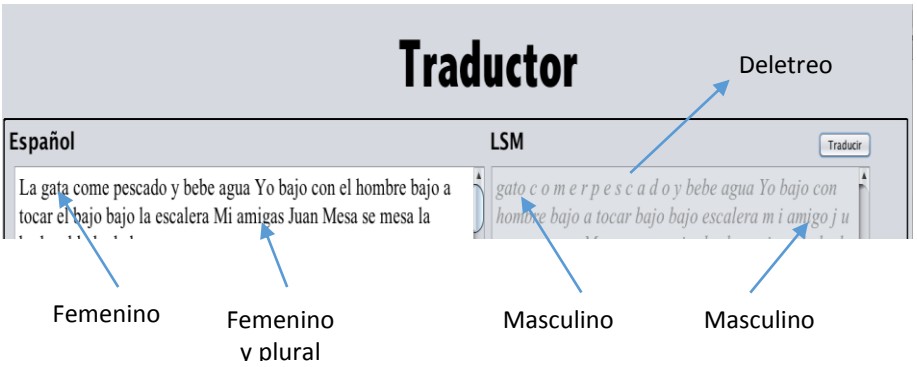

Fig. 6. Ejemplo de traducción variando género y número.

Como se muestra en la Fig. 6. Se ingresó una frase cualquiera al traductor, y esta se reescribió en la sección de LSM de la forma en que se diría, se puede apreciar como las palabras que no se han localizado en la base de datos por lo que se deletrease identifica por encontrarse en cursiva y separadas por un espacio, de este ejemplo se puede observar que aún falta léxico para una traducción completa de la frase en original,

también es perceptible que los nombres propios siempre serán deletreados, ya que no existe seña para estos. Las palabras completas significa que han sido encontradas en la base de datos, tal cual, aparecen o el lema de la palabra basado en las reglas básicas mencionadas anteriormente. Un punto a destacar es que si no se implementara la sección de agregar la seña de mujer para femeninos y la seña de muchos para plurales, como suele hacerse en LSM, se escribirían solo los singulares y masculinos de las palabras, este ejemplo sirve para mostrar sin la implementación de esto, más adelante se mostrarán ejemplos de la traducción correcta.

Otro aspecto a resaltar es, la división de frases, se pueden ingresar varias frases y estas son analizadas independientemente de las otras.

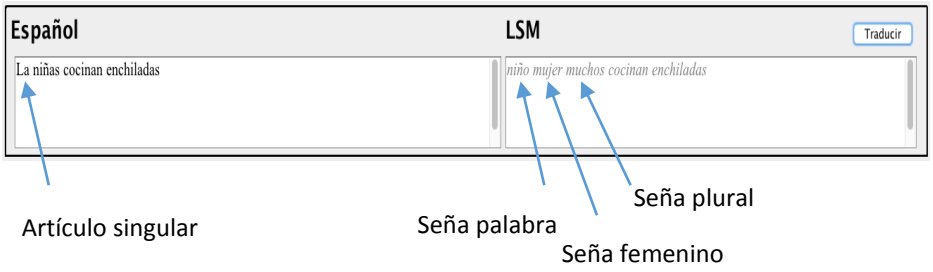

Fig. 7. Ejemplo de traducción de una frase con artículos.

En el ejemplo mostrado en la Fig. 7 se observa como al no realizarse un análisis semántico de las frases, se puede escribir "La" en lugar de "Las" y no marcará error, aunque al ser un artículo es descartado, la palabra "niñas" al ser un femenino y estar en plural, se escribe la seña del lema de la frase, más la seña de mujer, más la seña de muchos.

Fig. 8. Secuencia de video de una seña.

Como se menciona las señas son reproducidas de una secuencia de 10 imágenes, para reproducir un video de una frase se agrupan las secuencias de señas como se observa en la Fig. 8.

5 Conclusiones y trabajo a futuro

A pesar del esfuerzo que se está realizando por parte de algunas personas y asociaciones que tratan de difundir y mejorar la práctica de la lengua de señas, aún se encuentra con muchas dificultades para su enseñanza, ya que, no se cuenta con una gramática formal e intérpretes certificados. Por lo que en la actualidad la posibilidad de contar con un traductor, pueden brindar la posibilidad de conectar las realidades de dos culturas diferentes de una forma sencilla.

El traductor automático debería ser capaz de adecuar un mensaje expresado en una lengua origen a una lengua destino, para conseguir esto el sistema presentado tiene aspectos relevantes a mejorar que se ven limitados a la falta de normatividad en LSM. Actualmente realiza una la traducción directa del texto inscrito con algunas reglas, por lo cual, se debe seguir desarrollando e investigando más la gramática de la LSM, así como trabajar en la desambiguación. Como ejemplo claro de esto es la validación sintáctica de oraciones gramaticalmente correctas, ya que aún no se han incorporado los mecanismos para realizarlos, por lo que en algunas ocasiones las oraciones, carecen de sentido, y aun así realizará la traducción basado al etiquetado que realice Freeling.

El software no garantiza la adquisición de LSM, sin embargo es una herramienta útil para adquirir las bases para dicha lengua, contribuyendo así en gran medida a la educación y compresión de personas sordas. La herramienta es un buen primer acercamiento para descubrir un mundo que para la mayoría resulta totalmente nuevo y desconocido y fomentar a través del aprendizaje de la lengua, un cambio de actitud y mayor comunicación hacia las personas sordas.

Referencias

1. Alvarado, M.: Construcción de una pedagogía para la integración. Serie: Integración Normalizada en la Formación para el Trabajo: un proceso de inclusión social de la Organización Internacional del Trabajo. Montevideo, Uruguay (1998)
2. Cruz Aldrete, M.: Gramática de la Lengua de Señas Mexicana. Doctorado. Colegio de México, Centro de Estudios Lingüísticos y Literarios (2008)
3. Gelbukh, A.: Procesamiento de Lenguaje Natural y sus Aplicaciones. En Komputer Sapiens, Vol. 1, pp. 6–11 (2010)
4. Hutchins, W.J., Somers, H.L.: An introduction to machine translation, Vol. 362, London: Academic Press (1992)
5. Moroni, Moreno Ortiz, A.: Diseño e implementación de un lexicón computacional para lexicografía y traducción automática. Estudios de lingüística del español, 9, 000-0 (2000)
6. Gelbukh, A., Sidorov, G.: Procesamiento automático del español con enfoque en recursos léxicos grandes. Dirección de Publicaciones del IPN, México (2010)
7. Trujillo, I.A.: Lexicalist Machine Translation of Spatial Prepositions. Ph.D. Dissertation. Trinity Hall, University of Cambridge (1995)
8. Lipschutz, Eatough, Andy: Mexican Sign Language Grammar. (Manuscrito inédito) (1992)

9. Fridman Mintz, B.: Categorías verbales de aspecto y tiempo en la Lengua de Señas Mexicana. Lubbers Quesada, M., Maldonado, R. (Eds.). Dimensiones del aspecto en español, pp. 195–244, Universidad Nacional Autónoma de México. Universidad Autónoma de Querétaro (2005)
10. Hawayek, A.: El orden lineal de los objetos del verbo en la Lengua de Señas de México. Signos lingüísticos, Vol. 2, pp. 25–49, México: UAM-Iztapalapa (2005)
11. Treviño, E., Hawayek, A.: Stages in the development of grammatical space. Proceeding of the 23rd, Annual Boston University Conference on Language Development, Vol. 2, Cascadilla Press Somerville, Mass. (1999)
12. Union Nacional de Sordos. https://unsordosm.wordpress.com/lengua-de-senas/. Revisado 18 marzo 2016.
13. Pichardo-Lagunas, O., Martinez-Seis, B.: Resource Creation for Automatic Translation System from Texts in Spanish into Mexican Sign Language. Research in Computing Science, Vol. 100, pp. 129–137 (2015)
14. Radelli, B.: Una aplicación de la lingüística: la logogenia. Dimensión Antropológica, Vol. 23, septiembre-diciembre, 2001, pp. 51–72. Disponible en: http://www.dimensionantropologica.inah.gob.mx/?p=652. Revisado 18 marzo 2016

Importancia del lenguaje coloquial y de los símbolos de puntuación en el perfilado de autores

Diana M. Sepúlveda-Barrera[1], Daniel Martínez-Espino[1],
Esaú Villatoro-Tello[2], Gabriela Ramírez-de-la-Rosa[2]

[1] Universidad Autónoma Metropolitana (UAM) Unidad Cuajimalpa,
Maestría en Diseño, Información y Comunicación (MADIC),
División de Ciencias de la Comunicación y Diseño,
México

[2] Universidad Autónoma Metropolitana (UAM) Unidad Cuajimalpa,
Departamento de Tecnologías de la Información,
México

{dcg.disb,damaes83}@gmail.com, {evillatoro,gramirez}@correo.cua.uam.mx

Resumen. En años recientes el perfilado de autores (PA) se ha convertido en una tarea muy relevante para la comunidad de Procesamiento del Lenguaje Natural (PLN). El objetivo principal del PA es determinar de forma automática características demográficas del autor, por ejemplo género y edad. En este trabajo presentamos una propuesta para resolver el problema de PA; en particular nos interesó determinar el rol que juega el lenguaje coloquial así como el significado de los distintos símbolos de puntuación. Contrario a trabajos previos, nuestra propuesta considera cada símbolo de puntuación de manera independiente y no como un solo atributo que abarca todos los símbolos de puntuación. Nuestra hipótesis plantea que el uso de determinados símbolos de puntuación en conjunto con lenguaje coloquial aportan información relevante a un método de clasificación automática. Como contribución adicional de este trabajo, nos dimos a la tarea de compilar un diccionario de lenguaje coloquial, el cual consideramos un recurso valioso para la comunidad de PLN haciendo investigación en áreas afines. Los resultados obtenidos muestran que los atributos propuestos permiten enriquecer positivamente esquemas tradicionales de representación de textos.

Palabras clave: Perfilado de autores, atributos estilísticos, representación de textos, procesamiento de lenguaje natural, aprendizaje supervisado.

Importance of Colloquial Language and Punctuation for Author Profiling

Abstract. In recent years, author profiling (AP) has become a very relevant task for natural language processing (NLP). The main goal of

AP is automatically determine demographic aspects from an author, for example, genre and age. In this paper we present a method for author profiling; particularly, we are interested in determine the rol of colloquial language and the meaning of diverse punctuation marks. Contrary to previous works, our proposal considers each punctuation mark independently and not as a single attribute that covers all marks. Our hypothesis states that the use of certain punctuation marks together with the use of colloquial language can provide relevant information to a automatic classification method. As an aditional contribution, we compiled and made available a dictionary with the colloquial words we use in this paper. The obteined results show that the proposed features allow enhance traditional text representation schemas.

Keywords: Author profiling, stylistic features, text representation, natural language processing, supervised learning.

1. Introducción

El perfilado de autor es uno de los retos recientes que ha llamado la atención de la comunidad científica, en particular de áreas como el procesamiento de lenguaje natural, ciencias forenses, estrategias de marketing y seguridad en internet. El objetivo principal del perfilado de autor (PA) es distinguir, a partir de un texto, entre clases de autores y no identificar a un autor en particular, siendo este último el escenario del problema conocido como atribución de autoría [14]. Así entonces, la tarea de PA busca modelar a través de atributos sociolingüísticos más generales a grupos de autores, dichos atributos son además indicadores de cómo los distintos grupos de autores emplean el lenguaje dependiendo de su género, edad y/o lenguaje nativo [1].

Uno de los primeros trabajos que enfrentaron el problema de PA fueron los propuestos en [1,6], donde se mostró, a través de técnicas estadísticas, que el análisis sobre el uso de las palabras en distintos documentos permite determinar el género, edad, idioma nativo e incluso la personalidad del autor. A partir de entonces, muchos trabajos se han propuesto resolver el problema de PA, ejemplos de estas investigaciones son [12,3,10,9,8]. En muchos de estos trabajos se ha enfatizado el uso y análisis de representaciones textuales, las cuales han mostrado ser bastante eficientes cuando los documentos que se quieren clasificar son escritos formales (*e.g.,* artículos de noticias, libros, etc.). Sin embargo, cuando se trata de textos informales (*e.g.,* blogs, chats, tuits), las representaciones tradicionales tienen problemas determinando el perfil de los autores. Esto se debe en gran parte a la dificultad que representa analizar textos informales, los cuales contienen frecuentemente muchos errores ortográficos, variado uso de *jerga* específica de los medios sociales, así como el uso excesivo de emoticonos. A partir de esto, surge la idea de utilizar atributos estilísticos en combinación con representaciones tradicionales para resolver el problema de PA en medios sociales [12].

En este trabajo proponemos una representación enriquecida que contempla dos aspectos: *i)* el uso del lenguaje coloquial (*i.e.,* jerga) en los textos; y *ii)* el uso y significado de distintos símbolos de puntuación. Nuestra hipótesis plantea que la combinación del uso de lenguaje coloquial junto con el uso de distintos símbolos de puntuación, considerando la frecuencia de uso de ambos, podría ser un factor importante al momento de distinguir el perfil del autor. Agregado a esto, se hizo la compilación de un diccionario coloquial, el cual es el resultado de identificar la *jerga* más comúnmente empleada en distintos y variados medios sociales. Este diccionario es un recurso valioso para la comunidad de PLN haciendo investigación en el área de PA.

El resto de este documento se encuentra organizado de la siguiente manera. En la sección 2 se describen algunos de los trabajos relacionados al problema de perfilado de autor. En la sección 3 se describe en detalle el método empleado en el perfilado de autor, así como los atributos estilísticos propuestos. En la sección 4 se describe la metodología experimental y los resultados obtenidos con el método propuesto. Finalmente, en la sección 5 se plantean las conclusiones obtenidas y se describen algunas lineas de trabajo futuro.

2. Trabajo relacionado

El problema del perfilado de autor ha sido atractivo para diferentes áreas de conocimiento, por ejemplo la psicología, lingüística, socio-lingüística y el procesamiento del lenguaje natural [6,12]. Tradicionalmente, como se describe en [8], el perfilado de autor involucra: *i)* la identificación y extracción de atributos textuales, *ii)* la construcción de una representación apropiada, por ejemplo bolsa de palabras (BOW[3]), y *iii)* la construcción de un modelo de clasificación, el cual es entrenado para identificar los perfiles de interés (*e.g.,* género o edad).

Dado lo anterior, uno de los trabajos que busca hacer detección de edad en conversaciones en línea, con el objetivo de identificar depredadores sexuales es el descrito en [15]. La hipótesis de los autores plantea que si un sistema de PA es capaz de detectar cuando un usuario adulto se quiere hacer pasar por un niño o adolescente, el problema del acoso sexual en internet se podría ver disminuido. En dicho trabajo se utilizó la biblioteca de NLTK [7] junto con el corpus NPS Chat Corpus, datos que reúnen textos provenientes de diferentes servicios de conversaciones en línea. Para probar su método, los autores formaron seis diferentes clases: adolescentes (13-19), adultos (20-59), 20s, 30s, 40s y 50s. Los atributos empleados fueron el uso de n-gramas de palabras, emoticones, tokens de puntuación (signos de puntuación), longitud promedio de oración y la cantidad de palabras promedio por documento. En los resultados obtenidos, las pruebas con SVM fueron siempre mejores, principalmente en el problema de identificar entre adolescentes *vs.* adultos (*i.e.* problema binario).

Por otro lado, en [5] se realiza un análisis estadístico en blogs para identificar variaciones del lenguaje dependiendo de la edad y del género. Los resultados

[3] Por sus siglas en Inglés: *Bag-of-Words*

obtenidos se basaron en dos atributos independientes; el primero, es el uso de lenguaje coloquial, y el segundo, en la longitud promedio de las oraciones según los grupos edad y género. Las pruebas se realizaron con un conjunto de 20,000 blogs como corpus, según los resultados reportados, la detección de género fue más acertada que la de edad. En general, el trabajo descrito en [5] se desprende de muchas de las ideas planteadas en [1,6].

En el trabajo descrito en [16] se utilizan conteos de características léxicas, semánticas y sintácticas para generar un sistema de clasificación de dos fases, el cual clasifica primero el género y posteriormente la edad. Otro trabajo que se aproxima al método propuesto en este artículo es el descrito en [11], donde los autores crearon un diccionario de emoticones. En sus experimentos, los autores muestran que es posible distinguir el genero de los autores a través de contar el número de emoticones empleados.

Más recientemente, el trabajo descrito en [8] se enfoca en proponer una representación vectorial comprimida (*i.e.*, pocas dimensiones) para eliminar el problema de la alta dimensionalidad que significa el trabajar con representaciones tipo BOW. Para esto, los autores proponen la construcción de sub-perfiles, donde la idea principal es capturar los elementos más discriminativos a nivel de intra-perfiles. Al final esta nueva representación es empleada en el esquema general de PA para entrenar un clasificador que permita identificar género y edad.

A diferencia del trabajo descrito previamente, en esta investigación nos interesa evaluar la importancia que tiene el significado de los diferentes símbolos de puntuación empleados por los autores. En trabajos previos los símbolos de puntuación son considerados indistintamente, es decir como un sólo atributo, así entonces una coma (,) es tratada igual que un punto y coma (;) o que dos puntos (:), etc. Nuestra intuición es que el uso y significado que les dan los autores a estos símbolos en conjunto con el uso de lenguaje coloquial (*jerga*), pueden ser factores relevantes al momento de entrenar un clasificador para identificar el género y el rango de edad de los autores.

3. Método propuesto

3.1. Representación de los documentos

En este trabajo se aborda la problemática de la identificación del perfil de autor aplicando el paradigma de clasificación de textos (CT)[4]. Bajo este paradigma el primer paso consiste en el *indexado* de los documentos de entrenamiento (Tr), esta actividad consiste en mapear un documento d_j a una forma compacta de su contenido. La representación más comúnmente utilizada para representar cada documento es un vector con términos ponderados como entradas, concepto tomado del modelo de espacio vectorial usado en recuperación de información [2]. Es decir, un texto d_j es representado como el vector $\overrightarrow{d}_j = \langle w_{kj}, \ldots, w_{|\tau|j} \rangle$, donde

[4] La Clasificación de Textos es la tarea de asociar automáticamente categorías predefinidas con documentos a partir del análisis de su contenido [13].

τ es el *diccionario*, *i.e.*, el conjunto de términos que ocurren al menos una vez en algún documento de Tr, mientras que w_{kj} representa la importancia del término t_k dentro del contenido del documento d_j.

El peso w_{kj} se puede determinar de distintas maneras, entre las más usadas en la comunidad científica están el ponderado booleano y el ponderado por frecuencia relativa de términos. Una breve descripción es dada a continuación:

- *Ponderado Booleano:* Consiste en asignar el peso con valor de 1 si la palabra ocurre en el documento y 0 en otro caso:

$$w_{kj} = \begin{cases} 1, \text{ si } t_k \in d_j \\ 0, \text{ en otro caso.} \end{cases} \tag{1}$$

- *Ponderado por frecuencia relativa (TF-IDF):* Este tipo de ponderado es una variación del tipo anterior y se calcula de la siguiente forma:

$$w_{kj} = TF(t_k) \times IDF(t_k), \tag{2}$$

donde $TF(t_k)$ es la frecuencia del término t_k en el documento d_j. IDF es conocido como la "frecuencia inversa" del término t_k dentro del documento d_j. El valor de IDF es una manera de medir la "rareza" del término t_k. Para calcular el valor de IDF se utiliza la siguiente ecuación:

$$IDF(t_k) = \log \frac{|D|}{|\{d_j \in D : t_k \in d_j\}|}, \tag{3}$$

donde D representa la colección de documentos que está siendo indexada.

3.2. Atributos de estilo

Como estrategia para enriquecer la representación BOW se decidió considerar la presencia de 8 atributos estilísticos. A continuación se listan los atributos de estilo contemplados:

- LINK - indica la frecuencia de uso de URLs en los documentos en revisión
- EMOTICON - indica la frecuencia del uso de emoticonos, para esto se utilizó un diccionario de los emoticonos más comunes (Sección 3.3).
- IMG - representa la presencia de imágenes en los documentos.
- PUNTUATRESP - indica la frecuencia de uso de puntos suspensivos (...)
- PUNTUAP - refiere a la aparición de puntos separadores de oraciones y/o párrafos (.).
- PUNTUAC - indica la frecuencia de uso de comas (,).
- PUNTUADOSP - representa la aparición del símbolo de dos puntos (:).
- PUNTUAPC - indica la presencia del símbolo de punto y coma (;).

3.3. Diccionario de lenguaje coloquial

Como elemento estilístico adicional se definió el atributo INFORMALEXXX, el cual es un atributo que refleja la cantidad de lenguaje informal/coloquial (*jerga*) empleado por el autor. Para poder calcular este atributo fue necesaria la compilación de un diccionario de lenguaje informal.

Así entonces, el diccionario fue creado haciendo una recopilación de diversas listas de jerga digital en español e inglés (fail, noob, gamer, etc.), símbolos representativos de emoticones (n_n, :D, x_x, etc.), y siglas de uso común (lol, brb, afk, etc.), obtenidos de fuentes como Wikipedia, foros de uso popular como Taringa!, blogs latinoamericanos de análisis de tendencias digitales y de uso personal. El diccionario quedó conformado por 1,178 palabras y símbolos[5].

En la Tabla 1 se puede observar la incidencia en porcentaje de palabras contenidas en el diccionario coloquial digital que aparecen en los textos del corpus utilizado para nuestros experimentos. Como es posible observar, el porcentaje de uso de lenguaje coloquial es notoriamente diferente entre distintos rangos de edades, por lo cual se consideró para hacer las distinciones.

Tabla 1. Porcentaje de incidencia de palabras coloquiales en el corpus

Clase	Porcentaje de aparición
Hombres	2.83 %
Mujeres	2.02 %
10's (11-19)	9.48 %
20's (20-29)	4.55 %
30's (30-39)	2.01 %

3.4. Métodos de clasificación

Dado que nuestra propuesta para identificar perfiles de autores no depende en particular de ningún algoritmo de aprendizaje, podemos emplear prácticamente cualquier clasificador para enfrentar el problema. Para los experimentos realizados seleccionamos dos diferentes algoritmos de aprendizaje, los cuales son algoritmos representativos dentro de la gran variedad de algoritmos de aprendizaje disponibles actualmente en el campo de aprendizaje computacional. Específicamente, consideramos los siguientes:

- **Naïve Bayes(NB).** Método probabilístico que asume la independencia de los atributos entre las diferentes clases del conjunto de entrenamiento.
- **Arboles de decisión (J48)**. Un algoritmo que permite generar un árbol de decisión, el cual selecciona los atributos más discriminativos basándose en su medida de entropía.

[5] El recurso está disponible en: `http://ccd.cua.uam.mx/~evillatoro/Resources/Slang_Dictionary_RCS_2016.txt`

En nuestros experimentos se empleó la implementación de Weka de cada uno de estos algoritmos, donde los parámetros empleados fueron los entregados por defecto por Weka [4]. Es importante mencionar que para todos los experimentos se aplicó como estrategia de validación la técnica de validación cruzada a diez pliegues.

3.5. Evaluación

Para evaluar un sistema de clasificación se utilizan las medidas de *Precisión* y *Recuerdo*, que son medidas comunes en el área de recuperación de información. La precisión (P) es la proporción de documentos clasificados correctamente en una clase c_i con respecto a la cantidad de documentos clasificados en esa misma clase. El recuerdo (R), la proporción de documentos clasificados correctamente en una clase c_i con respecto a la cantidad de documentos que realmente pertenecen a esa clase. Así, la precisión se puede ver como una medida de la corrección del sistema, mientras que el recuerdo da una medida de cobertura o completitud.

Adicionalmente, es común emplear la medida-F para describir el comportamiento de la clasificación, la cual se define como:

$$medida - F = \frac{(1+\beta^2)Precision * Recuerdo}{\beta^2 Precision + Recuerdo}, \tag{4}$$

donde con $\beta = 1$ representa la media armónica entre la precisión y el recuerdo. La función de β es la de controlar la importancia relativa entre las medidas de precisión y recuerdo. Es común asignar un valor de 1 indicando igual importancia a ambas medidas.

4. Resultados experimentales

4.1. Conjunto de datos

Para la realización de nuestros experimentos trabajamos con datos extraídos del corpus PAN-2013[6], el cual está compuesto por documentos en inglés y en español. Nuestro trabajo se enfocó a realizar las pruebas con los textos en español. El corpus contiene documentos de 75,900 autores diferentes, ambos géneros y distintos segmentos de edad (37,950 documentos para cada género; 2,500 textos en el segmento de 11 a 19 años de edad, que denominaremos 10's; 42,600 textos en el segmento de 21 a 29 años de edad, que denominaremos 20's; 30,800 documentos en el segmento de 31 a 39 años de edad, que llamaremos 30's).

En general, los documentos que conforman el corpus pertenecen a distintas tipologías de texto, como entradas de blogs, mensajes de foros, conversaciones de chat, anuncios, noticias, artículos, por mencionar algunos, en donde todos los autores tienen como enlace común el idioma (incluso pertenecen a distintas

[6] `http://pan.webis.de/clef13/pan13-web/author-profiling.html`

nacionalidades). Para realizar nuestros experimentos, fue necesario reducir la muestra de datos, esto principalmente debido a las limitaciones en cuanto a poder de cómputo se refiere. En la Tabla 2 podemos observar como quedó conformado el corpus final. En la tabla se muestra información referente al número de documentos por clase (Num. documentos), el tamaño promedio de cada documento (medido en tokens), tamaño promedio de tokens (medido en caracteres); tamaño promedio de oraciones (cantidad de tokens por oracion); y diversidad léxica (número de veces promedio que se utilizan una palabra en todo el documento).

Tabla 2. Estadísticas de los documentos en el corpus empleado

	Género		*Edad*		
	Hombres	Mujeres	10's	20's	30's
Num. documentos	300	300	200	200	200
Tamaño documentos	1984.05	1840.38	1309.09	2171.74	1677.87
Tamaño tokens	5.52	5.50	5.82	5.48	5.54
Tamaño oraciones	98.66	96.56	79.92	95.34	92.28
Diversidad léxica	1.35	1.35	1.28	1.40	1.35

4.2. Resultados del método base

Como se mencionó en secciones anteriores, como método base se empleó una forma de representación de bolsa de palabras. Bajo este esquema, la dimensionalidad del vector de atributos es de 19,275 atributos. La Tabla 3 muestra el desempeño obtenido para los problemas de identificación de género y edad respectivamente.

Tabla 3. Resultados de clasificación de género y edad empleando una bolsa de palabras

Pesado - Algoritmo	*Género*			*Edad*		
	Precisión	*Recuerdo*	*F-score*	*Precisión*	*Recuerdo*	*F-score*
BOOL-NB	**0.571**	**0.567**	**0.56**	**0.442**	0.415	0.39
BOOL-J48	0.549	0.548	0.54	0.419	**0.422**	**0.42**
TF-IDF-NB	0.554	0.553	0.55	0.412	0.410	0.40
TF-IDF-J48	0.534	0.533	0.53	0.381	0.383	0.38

Se puede observar que en general un esquema de pesado booleano permite a los clasificadores obtener un mejor desempeño en términos de *precisión*, lo cual significa que la presencia/ausencia de ciertos términos es un factor importante al resolver el problema de perfilado.

Al hacer un análisis sobre los atributos con mayor ganancia de información para el problema de identificación de género, observamos que los atributos más relevantes contienen una carga emocional, por ejemplo: *Sonrisa, reales, diez, ley, familias, tierras, inlove, dy, letras, lamento*. Por otro lado, para el caso de la identificación de edad los atributos con mayor ganancia de información fueron términos asociados a la jerga empleada en medio sociales, por ejemplo: *k, hora, fr, go, qe, qu, errores, contar, super, spero*. A partir de este análisis, se decidió realizar un experimento adicional, el cual consistió en aplicar como técnica de reducción de dimensionalidad la estrategia de ganancia de información. Tras este proceso, se realizaron nuevamente los experimentos empleando el método base más ganancia de información (IG). Los resultados obtenidos bajo este esquema aparecen en la Tabla 4

Tabla 4. Resultados de clasificación de género y edad empleando una bolsa de palabras con ganancia de información

Pesado - Algoritmo	*Género*			*Edad*		
	Precisión	*Recuerdo*	*F-score*	*Precisión*	*Recuerdo*	*F-score*
BOOL-NB	0.678	**0.645**	**0.63**	0.543	**0.498**	**0.48**
BOOL-J48	0.589	0.555	0.51	0.474	0.445	0.42
TF-IDF-NB	**0.729**	0.597	0.53	0.532	0.428	0.40
TF-IDF-J48	0.644	0.532	0.42	**0.562**	0.403	0.33

Como es posible observar, la reducción de dimensionalidad a través de IG permite al clasificador mejorar significativamente su desempeño en términos de la medida F. Es particularmente notorio el desempeño de un esquema de pesado booleano empleando un clasificador bayesiano (BOOL-NB).

4.3. Resultados del método propuesto

En los siguientes experimentos se reportan los resultados obtenidos al hacer la extensión de la BOW por medio de incorporar los atributos estilísticos definidos en la sección 3, los cuales capturan las frecuencias de uso de emoticones, links, diferentes signos de puntuación y vocabulario coloquial digital.

Los resultados de los experimentos realizados en esta sección están descritos en la Tabla 5 y Tabla 6 para cuando no se aplica ganancia de información y cuando si se hace uso de IG respectivamente.

Como se puede observar, la representación propuesta, sin emplear ganancia de información, funcionó mejor para el caso de identificación de edad. Nótese que para el caso de género, los resultados son muy similares a los obtenidos con el método base (Tabla 3). Por el contrario, para el problema de identificación de edad, se pudo lograr una pequeña mejora en los resultados, de aproximadamente un 1 % relativo (Tabla 3 *vs.* Tabla 5).

Tabla 5. Resultados de clasificación de género y edad empleando una bolsa de palabras extendida con los atributos estilísticos propuestos

Pesado - Algoritmo	*Género*			*Edad*		
	Precisión	*Recuerdo*	*F-score*	*Precisión*	*Recuerdo*	*F-score*
BOOL-NB	**0.573**	**0.568**	**0.56**	**0.446**	0.418	0.39
BOOL-J48	0.533	0.533	0.53	0.422	0.423	0.42
TF-IDF-NB	0.540	0.540	0.53	0.436	**0.433**	**0.43**
TF-IDF-J48	0.543	0.543	0.54	0.399	0.402	0.39

Tabla 6. Resultados de clasificación de género y edad empleando una bolsa de palabras extendida con los atributos estilísticos propuestos y utilizando ganancia de información

Pesado - Algoritmo	*Género*			*Edad*		
	Precisión	*Recuerdo*	*F-score*	*Precisión*	*Recuerdo*	*F-score*
BOOL-NB	0.703	**0.698**	**0.70**	0.440	0.440	0.43
BOOL-J48	0.606	0.605	0.60	0.457	0.448	0.44
TF-IDF-NB	**0.773**	0.600	0.53	**0.557**	**0.463**	**0.44**
TF-IDF-J48	0.737	0.600	0.53	0.473	0.460	0.44

A pesar de la mejora mínima en el desempeño del clasificador para el caso de identificación de edad, un análisis sobre los atributos con mayor ganancia de información arrojó que entre los atributos con mayor relevancia están los siguientes: *k, hora, PUNTUAC, fr, go, qe, qu, PUNTUATRESP, INFORMALEXXX, errores, EMOTICON*. De manera similar, para el caso de identificación de género, los atributos con mayor ganancia de información fueron: *LINK, sonrisa, diez, reales, EMOTICON, ley, familias, dy, canto y lamento*. Este análisis indica, hasta cierto punto, que los atributos propuestos para enriquecer la bolsa de palabras son efectivamente discriminativos en ambos problemas. Dado que los atributos propuestos aparecieron entre los atributos con mayor IG, se replicaron los experimentos empleando la bolsa de palabras enriquecida aplicando IG como estrategia de reducción de dimensionalidad. Los resultados obtenidos se muestran en la Tabla 6.

Como es posible observar, para el caso de identificación de género se logró un 70 % en la medida F, lo cual representa una ganancia significativa en comparación con emplear la BOW enriquecida sin aplicar IG (Tabla 5). Para el caso de identificación de edad es posible obtener mejoras mínimas en general.

Como pruebas adicionales se realizó una serie de experimentos en los cuales no se hizo distinción entre los diferentes símbolos de puntuación, es decir, se consideraron los atributos descritos en la sección 3 como un solo atributo en el proceso de representación de los datos[7]. Así entonces, para la tarea de identifica-

[7] Esta representación corresponde a lo que normalmente se ha aplicado en trabajos previos (Ver sección 2)

ción de género, se replicó la mejor configuración de la Tabla 6 (*i.e.*, BOOL-NB), y los resultados obtenidos fueron: $F = 0.48$, $P = 0.53$ y $R = 0.50$. Note que el desempeño decrece en comparación con los resultados de la Tabla 6, lo cual muestra la pertinencia de hacer la distinción entre símbolos de puntuación para el problema de identificación de *género*. De manera similar, para el problema de identificación de edad se tomó la mejor configuración lograda de la Tabla 6 (*i.e.*, TF-IDF-NB), para este caso los resultados fueron: $F = 0.40$, $P = 0.41$ y $R = 0.40$. Nuevamente, el desempeño del clasificador empeora, evidenciando la efectividad de hacer una caracterización individualizada de los símbolos de puntuación.

4.4. Discusión

Como se mencionó en la sección anterior, los atributos estilísticos propuestos para enriquecer la representación BOW mostraron tener mayor pertinencia en el problema de identificación de edad. Técnicas de selección de atributos (*e.g.,* IG) identifican como características relevantes, *i.e.,* asignan un *rank* alto, a un subconjunto importante de los atributos propuestos. A continuación (Tabla 7) mostramos un ejemplo que refleja el uso distinto que diferentes autores, de diferentes rangos de edad, dan a los símbolos de puntuación en sus textos.

De estos ejemplos se puede notar una tendencia hacia el uso indiscriminado de símbolos de puntuación entre más joven es el autor (*e.g.* autores del rango 10's). Por el contrario entre mayor edad tiene el autor, tiende a hacer un uso más ordenado y correcto de dichos símbolos, además de que hace uso de diversos símbolos y no unos cuantos, que es el caso de los mas jóvenes.

En general, el análisis realizado indica que el hacer distinción entre los símbolos de puntuación empleados es un elemento relevante para sistemas de PA. Contrario a trabajos anteriores, nosotros consideramos que hacer una distinción explícita de los distintos símbolos de puntuación es importante debido a que distintos autores tienden a utilizarlos de formas distintas, tal y como se muestra en la Tabla 7. Agregado a esto, la secuencia en que aparecen los mismos podría ser otro factor atractivo a considerar como un atributo adicional en un sistema de perfilado de autor.

5. Conclusiones

En este trabajo hemos descrito nuestro método para enfrentar el problema del perfilado de autor. Nuestro objetivo principal fue determinar la pertinencia del uso del lenguaje coloquial *digital* en conjunto con distintos símbolos de puntuación para determinar el perfil de autor. Contrario al trabajo previo, nosotros hacemos una distinción entre símbolos de puntuación, bajo el supuesto de que varios autores los utilizan con un significado, forma y frecuencias diferentes.

Para la realización de nuestros experimentos se tomó una muestra aleatoria de datos extraídos de la competencia del PAN-2013. Los documentos de dicha competencia son en su mayoría datos obtenidos de medios sociales, lo cual

Tabla 7. Porciones de textos extraídos del corpus utilizado que reflejan la diversidad de uso que se les da a los distintos símbolos de puntación

Muestra de texto de documento perteneciente a clase 10's
niñas lo mejor que me ha pasado en la vida juntas por siempre nenas sois mi vida enteraa os adoroo encerioo no cambieis cada una de vosotras es unica de difernte maneraaa! oOOOooOOoOOooO PUNTUADOSP
Muestra de texto de documento perteneciente a clase 20's
Me convertí en adulto y hasta hoy cuando veo a una persona hablando demasiado PUNTUAC interrumpiendo la conversación de todos PUNTUAC siendo inoportuna o violenta PUNTUAC presumiendo de lo que tiene PUNTUAC sintiéndose prepotente y haciendo de menos a la gente PUNTUAC tengo la impresión de oír la voz de mi padre diciendo PUNTUADOSP "Cuanto más vacía la carreta PUNTUAC mayor es el ruido que hace"
Muestra de texto de documento perteneciente a clase 30's
Quizás sea la rama poética la que más siente el gallego-berciano PUNTUAP Pero también hay que comprender que los poetas utilizan el código escrito que más dominan PUNTUAC el castellano PUNTUAC porque es el que recibieron durante su formación educativa PUNTUAP Sólo los poetas más populares PUNTUAC en el sentido de cercanos a su tierra PUNTUAC se atreven con el gallego PUNTUAP Este es el caso paradigmático de Gilberto Núñez Ursinos PUNTUAC tan concienciado por la poesía social PUNTUAC "déixame tamén a min escuitar PUNTUAC oilo aturuxo os teus fillos que ben cedo emigraron" PUNTUAP Por su parte PUNTUAC el poeta Antonio Bouza Pol declara con firmeza que "el gallego es nuestra segunda lengua PUNTUAC y esta realidad antiquísima tiene que ser conocida PUNTUAC respetada y querida por toda la Comunidad de Castilla y PUNTUAC más aún PUNTUAC por León"

representa un reto importante para los sistemas automáticos de PA debido a la presciencia de errores ortográficos, el uso frecuente de jerga específica de medios sociales, así como el uso de emoticones. Los resultados obtenidos de nuestros experimentos muestran que el método propuesto es pertinente para el problema de identificación de edad.

Un análisis realizado de ganancia de información sobre el vector de atributos reflejó que 5 de los 9 atributos estilísticos propuestos aparecen entre los atributos más relevantes para resolver el problema de PA. Experimentos realizados muestran que, tras hacer una reducción de dimensionalidad empleando la estrategia de IG, el desempeño del clasificador mejora significativamente. Estos resultados nos motivan a seguir trabajando sobre esta línea de investigación. Agregado a esto, como una contribución adicional de nuestro trabajo se conformó, a través

de distintas fuentes, un diccionario de lenguaje coloquial digital, el cual contiene una gran variedad de términos de la *jerga* más comúnmente empleada en medios sociales. Este recurso lingüístico lo consideramos de gran importancia para la comunidad científica de PLN trabajando en áreas afines.

Como trabajo futuro nos proponemos hacer pruebas con una muestra mayor, de forma que podamos validar los hallazgos hasta ahora encontrados. También planeamos hacer una representación que considere la secuencia de aparición de los símbolos de puntuación, por ejemplo n-gramas de símbolos de puntuación. Finalmente, una definición más fina de atributos asociados a símbolos de puntuación podría resultar en un beneficio mayor del sistema de clasificación.

Agradecimientos. Este trabajo fue parcialmente financiado por el CONACyT a través de las becas 708534 y 717783, el proyecto de investigación No. 258588 y programa del SNI. También se agradece el apoyo otorgado a través de la Coordinación de la Maestría en Diseño, Información y Comunicación (MADIC) de la UAM Cuajimalpa.

Referencias

1. Argamon, S., Koppel, M., Fine, J., Shimoni, A.R.: Gender, genre, and writing style in formal written texts. TEXT 23, 321–346 (2003)
2. Baeza-Yates, R., Ribeiro-Neto, B., et al.: Modern information retrieval, vol. 463. ACM press New York (1999)
3. Burger, J.D., Henderson, J., Kim, G., Zarrella, G.: Discriminating gender on twitter. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 1301–1309. EMNLP '11, Association for Computational Linguistics, Stroudsburg, PA, USA (2011), http://dl.acm.org/citation.cfm?id=2145432.2145568
4. Garner, S.R.: Weka: The waikato environment for knowledge analysis. In: In Proc. of the New Zealand Computer Science Research Students Conference. pp. 57–64 (1995)
5. Goswami, S., Sarkar, S., Rustagi, M.: Stylometric analysis of bloggers' age and gender. In: Third International AAAI Conference on Weblogs and Social Media (2009)
6. Koppel, M., Argamon, S., Shimoni, A.R.: Automatically categorizing written texts by author gender. Literary and Linguistic Computing 17(4), 401–412 (2002), http://llc.oxfordjournals.org/content/17/4/401.abstract
7. Loper, E., Bird, S.: Nltk: The natural language toolkit. In: Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1. pp. 63–70. ETMTNLP '02, Association for Computational Linguistics, Stroudsburg, PA, USA (2002), http://dx.doi.org/10.3115/1118108.1118117
8. López-Monroy, A.P., y Gómez, M.M., Escalante, H.J., Villaseñor-Pineda, L., Stamatatos, E.: Discriminative subprofile-specific representations for author profiling in social media. Knowledge-Based Systems 89, 134 – 147 (2015), http://www.sciencedirect.com/science/article/pii/S0950705115002427

9. Nguyen, D., Gravel, R., Trieschnigg, D., Meder, T.: How old do you think i am?; a study of language and age in twitter. In: Proceedings of the Seventh International AAAI Conference on Weblogs and Social Media. AAAI Press (2013), reporting year: 2013
10. Peersman, C., Daelemans, W., Van Vaerenbergh, L.: Predicting age and gender in online social networks. In: Proceedings of the 3rd International Workshop on Search and Mining User-generated Contents. pp. 37–44. SMUC '11, ACM, New York, NY, USA (2011), http://doi.acm.org/10.1145/2065023.2065035
11. Rangel, F.: Author profile in social media: Identifying information about gender, age, emotions and beyond. In Proceedings of the 5th BCS IRSG Symposium on Future Directions in Information Access pp. 58–60 (2013), http://ewic.bcs.org/upload/pdf/ewic_fdia13_paper14.pdf
12. Schler, J., Koppel, M., Argamon, S., Pennebaker, J.W.: Effects of age and gender on blogging. In: AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs. vol. 6, pp. 199–205 (2006)
13. Sebastiani, F.: Machine learning in automated text categorization. ACM Comput. Surv. 34(1), 1–47 (Mar 2002), http://doi.acm.org/10.1145/505282.505283
14. Stamatatos, E.: A survey of modern authorship attribution methods. J. Am. Soc. Inf. Sci. Technol. 60(3), 538–556 (Mar 2009), http://dx.doi.org/10.1002/asi.v60:3
15. Tam, J., Martell, C.H.: Age detection in chat. In: Semantic Computing, 2009. ICSC '09. IEEE International Conference on. pp. 33–39 (Sept 2009)
16. Yuridiana Alemán, D.V., Pinto, D.: Una metodología para la detección del perfil de un autor. Avances en la Ingeniería del Lenguaje y del Conocimiento 93 (2014), http://www.sciencedirect.com/science/article/pii/S0950705115002427

Determinación del género de autores de textos cortos a través de n-gramas

Francisco Antonio Castillo Velásquez, María Del Consuelo Patricia Torres Falcón, Ely Karina Anaya Rivera, Iván Peredo Valderrama, Jonny Paul Zavala de Paz

Universidad Politécnica de Querétaro,
Querétaro, México

{francisco.castillo, consuelo.torres, karina.anaya, ivan.peredo}@upq.mx,
jonny.zavala@upq.edu.mx

Resumen. En la actualidad, la posibilidad de comunicarse o de expresarse por un medio electrónico es muy amplia: correo electrónico, redes sociales, chats y otras herramientas son usadas por la mayoría de los usuarios de computadoras y dispositivos móviles. Uno de los problemas que se ha presentado con esta forma de comunicación es el exceso, como el plagio, falsa identidad, notas intimidatorias, etc. La atribución de autoría de textos (AAT) se encarga de responder a la cuestión de quién es el autor de un texto, dando algunos ejemplos previos de ese autor (conjunto de entrenamiento). Un proceso útil dentro de la AAT es la identificación de género o sexo (hombre, mujer) y que ha sido estudiado por varios autores pero principalmente para el inglés. El presente trabajo propone un modelo computacional basado en características léxicas (n-gramas) para la identificación del género para textos cortos en español. Se hicieron pruebas con un corpus de textos de mensajes en redes sociales y blogs, obteniendo resultados prometedores.

Palabras clave: Identificación de género, aprendizaje automático, n-gramas, clasificación, autoría

Gender Determination of Authors of Short Texts using N-grams

Abstract. Nowadays, the possibilities for communicating or expressing through an electronic way are very wide: e-mail, social networks, chats and other ways are used by the majority of computer and mobile device users. One of the problemas that is presented in this communication way is excess, such as plagiarism, identitiy falsification, blackmailing, etc. Text authorship attribution (TAA) is in charge of answering authoring issues by providing previous examples from said author (training set). A useful process within TAA is sex or gender identification (male, female), which has been studied by many authors for its use in English mostly. The present work proposes a computational model

based on lexical characteristics (n-grams) for gender identification in short texts in Spanish. Tests were carried out with a corpus from social network and blog text messages, producing promising results.

Keywords: Gender identification, machine learning, n-grams, classification, authorship.

1. Introducción

Las diferencias en la manera de expresarse verbalmente entre hombres y mujeres provienen de múltiples factores. Por una parte se encuentran elementos extrínsecos como la cultura, las exigencias sociales o la educación; y por otro lado, existen factores intrínsecos como las capacidades personales, el entrenamiento y la propia personalidad. De acuerdo con algunos estudios la manera de comunicarse en gran parte está determinada por las "diferencias en el funcionamiento y la estructura cerebral entre hombres y mujeres" [5]. Estas diferencias se manifiestan tanto en la comunicación oral como en la escritura.

La información textual que proporcionan los usuarios en redes sociales, sistemas de correo electrónico o blogs ofrece un potencial mercadológico y de seguridad, principalmente. Pero es indudable que parte de esa información no es del todo confiable. Muchos usuarios mienten sobre su edad, género, afiliación o gustos, apoyándose de fenómenos lingüísticos como el sarcasmo o la ironía; en otros casos, simplemente no reportan dicha información. Conocer el perfil demográfico o psicológico de tales usuarios es una oportunidad para las organizaciones y empresas y un reto para las tecnologías de Procesamiento del Lenguaje Natural (PLN).

Conocer el género del autor de un texto puede ser útil en tareas de la lingüística forense, como la identificación de escritos intimidatorios, detección de plagio y atribución de autoría.

Muchos estudios hechos están dirigidos a resolver el problema del género, pero la mayoría de estas investigaciones están limitadas al inglés y a los medios tradicionales. Este trabajo propone un modelo dirigido a texto en español (y con facilidad de aplicarse a muchos lenguajes) de fuentes como redes sociales y blogs.

Esta investigación presenta un modelo simple de identificación de género basado en la técnica de n-gramas, los cuales son secuencias de n elementos, que para el caso de la comunicación escrita pueden ser caracteres o palabras, por mencionar algunos. El principal objetivo es proporcionar un enfoque simple de identificación de género que pueda utilizarse para el español u otros lenguajes y con un grado de confiabilidad por lo menos similar a otros estudios hechos con otros enfoques diferentes.

El resto del artículo está organizado como sigue: en la sección 2 mencionamos un breve estado del arte de los estudios de la atribución o categorización del género. En la sección 3, presentamos los datos usados en este trabajo y definimos el problema de predicción del género. Las técnicas que explotamos para resolver el problema planteado se presentan en la sección 3. La sección 4 muestra los resultados de los experimentos llevados a cabo para evaluar la viabilidad de la predicción de género. Finalizamos discutiendo algunas conclusiones del efecto del género sobre el estilo de la escritura.

2. Estado del arte

Los estudios de autoría en la literatura pueden dividirse en tres categorías: atribución de autoría, detección de similitud y caracterización de la autoría. La atribución es la tarea de encontrar o validar el autor de un documento. Algunos ejemplos bien conocidos sobre atribución son la revisión de los trabajos de Shakespeare [4, 5] y la identificación de los autores de los disputados Documentos Federales (Federalist papers) [6, 7, 8]. La detección de similitud intenta encontrar la variación entre los escritos de un autor o diferenciar entre segmentos de texto escritos por diferentes autores, mayormente con propósitos de detección de plagio.

La caracterización de la autoría puede definirse como la tarea de asignar los escritos de un autor a un conjunto de categorías de acuerdo a un perfil sociolingüístico. Algunos atributos analizados previamente en la literatura son el género, nivel de educación, idioma y antecedentes culturales. En [11], se examina el género y el idioma usando técnicas de aprendizaje automático. En [12] se clasifican documentos en inglés de acuerdo al género del autor y al género del documento.

El análisis estilométrico también proporciona resultados interesantes, En general, las mujeres tienden a preferir usar palabras más grandes y con significado claro. A diferencia de los hombres, prefieren organizar oraciones más cortas y a omitir stopwords y signos de puntuación. El uso de caritas (*smiles*) y palabras que conlleven emociones es más común en las mujeres. Los mensajes largos de chat y el uso de palabras cortas son las características estilísticas más representativas de los hombres [7].

Procesando grandes cantidades de texto, usando un enfoque más orientado a palabras funcionales que en las de contenido y apoyándose de un software de análisis, Newman [10] intentó dar respuesta empírica a las cuestiones de cómo o por qué los hombres y las mujeres usan el lenguaje de forma diferente.

Yan & Yan [18] usaron la clasificación con Naïve Bayes para identificar el género de autores de blogs. Además de usar las características tradicionales de categorización, usaron algunas específicas, como los colores de fondo, fuente del texto y emoticones. Realizaron experimentos también con el enfoque de unigramas de palabras. Su corpus fue de 75000 entradas de 3000 bloggers. Sus experimentos más prometedores alcanzaron una precisión del 70%.

Chao-Yue [4] también desarrolló un clasificador Naïve Bayes y lo entrenó con frecuencias de palabras como principal característica. Sus resultados tuvieron una leve mejora con la adición de bigramas, trigramas y etiquetas PoS frecuentes. Su trabajo tomó en cuenta el efecto positivo que tienen las frases orientadas a la relación y las palabras orientadas al tópico sobre la precisión en los resultados, por lo que también realizó experimentos en donde las excluía.

Otro modelo de inferencia de género para redes sociales fue propuesto por Kokkos & Tzouramanis [6]. Ellos idearon una estrategia que explotaba tanto características basadas en contenido (características psicolingüísticas como los sentimientos no placenteros –ira, depresión, confusión, miedo) como características tradicionales de estilo (características basadas en carácter, en palabras y en sintaxis –total de palabras, cantidad de letras mayúsculas, cantidad de signos de interrogación, total de pronombres). La implementación del modelo incluyó un módulo de minería de texto

que combinó un etiquetador PoS y un clasificador SVM. Ellos reportan una precisión superior al 90% para pruebas con textos tomados de las redes Twitter y LinkedIn.

Nuestro trabajo de investigación está basado en un enfoque de n-gramas. De un estudio previo de nuestra parte surgió el concepto de n-gramas sintácticos. Estos son n-gramas definidos mediante caminos de un árbol sintáctico de dependencias o de constituyentes en lugar de la estructura lineal del texto [10]. Por ejemplo, la oración "las noticias económicas tienen poco efecto sobre los mercados financieros" puede ser transformada a n-gramas sintácticos siguiendo la estructura de sus relaciones de dependencia: *tienen-noticias*, *efecto-poco*, *tienen-sobre-mercados-los*.

Este tipo de n-gramas están destinados a reflejar la estructura sintáctica más fielmente que los n-gramas lineales, y tienen muchas de las mismas aplicaciones, especialmente como características en un Modelo de Espacio Vectorial. Los n-gramas sintácticos dan mejores resultados que el uso de n-gramas estándar para ciertas tareas, por ejemplo, para atribución de autoría [16]. En el presente trabajo no se usan n-gramas sintácticos, pero sí n-gramas simples de carácter, obteniendo resultados alentadores en la tarea de identificación de genero de autores.

3. Desarrollo

En muchas tareas del PLN los documentos son representados como vectores de características. Estos vectores pueden servir como entrada a varios algoritmos como de clusterización y clasificación de documentos. Las características más usadas son las léxicas y las de carácter, que consideran a un texto como una secuencia de palabras y de caracteres, respectivamente. La frecuencia de palabras, riqueza del vocabulario, n-gramas, frecuencias de letras, n-gramas de caracteres, etc., son ejemplos específicos. Una gran ventaja de estas características de bajo nivel es que son muy fáciles de extraer de forma automática [3].

Este trabajo también se vio motivado por la conclusión de Sarawgi [12], donde concluye que el enfoque más robusto está basado en modelos del lenguaje basados en caracter (que aprenden patrones morfológicos) más que en modelos basados en tokens (que aprenden patrones léxico-sintácticos).

En esta sección describiremos el modelo propuesto de trabajo, desde la compilación del corpus de trabajo, pasando por el proceso de obtención de los n-gramas y generación de estadísticas, hasta la tarea de clasificación (ver figura 1).

De forma general el proceso inicia con una recopilación de textos cortos que eventualmente formarán el corpus de trabajo. Estos textos son procesados para obtener n-gramas a nivel de carácter con un programa especial (text2ngram). Este programa no permite la ordenación por el campo de frecuencia por lo que se hizo necesario tener un proceso semi-automatizado para ordenar los n-gramas por frecuencias. El resultado de esta ordenación y de la generación del corpus son dos archivos de texto, los cuales son procesados para generar un archivo en formato reconocido por Weka, software que más adelante nos permitirá realizar la clasificación. Este archivo .arff contiene la información de las características representativas para la clasificación de género, que estarán definidas por los n-gramas más frecuentes. Se utilizaron varios algoritmos para el proceso de clasificación, entre ellos Naïve Bayes, máquina de soporte a vectores y

árboles de decisión. Finalmente, se obtienen las cifras de la clasificación. Estas etapas del modelo propuesto se detallan a continuación.

3.1 Compilación del corpus

La parte inicial del trabajo fue la generación de un corpus de mensajes de texto cortos en español (que no sobrepasaran de 300 caracteres) obtenidos de comentarios en diversas páginas que tenían añadido el plug-in de comentarios de Facebook. Este corpus estará disponible para la comunidad investigadora.

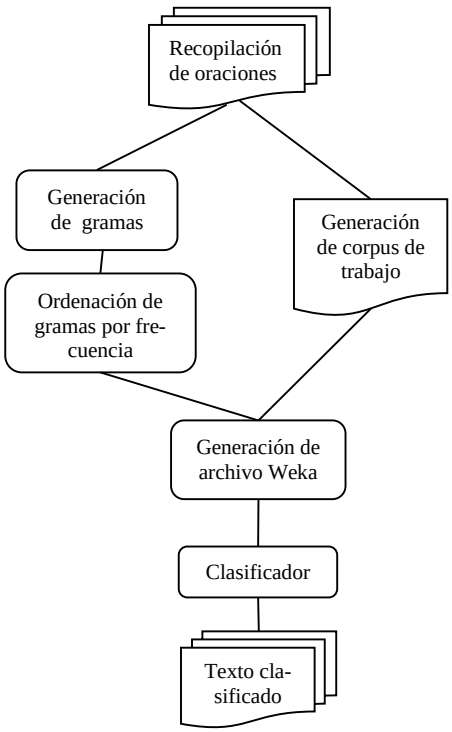

Fig. 1. Modelo propuesto para la identificación de género de textos cortos.

Se generó un corpus de 400 textos, la mitad de mujeres y la otra mitad de hombres. Como se mencionó anteriormente, cada uno de los textos no sobrepasa de 300 caracteres. A continuación se muestran algunos ejemplos de textos que forman parte del corpus. Se ha dejado la redacción original, inclusive con errores ortográficos o uso de *smyles*.

si lo sabes conservar sin caer en la monotonía sii es posible...!!
Ánimo compi no tengas meyo jeje pronto te iré a visitarte
No es gusto.. es por amor!
Extracto del corpus de textos de hombres

se pasa bien, estamos felices y mas unidos k nunk...

Guapaa!!! Ame tu falda
Ultimo día en mi trabajo :(pero que bonito detalle !!!!!!!
Extracto del corpus de textos de mujeres

Estos textos fueron compilados en un solo archivo plano con el campo del género y el texto correspondiente. Por ejemplo, los textos de los corpus anteriores quedarían compilados de la siguiente manera:

hombre,'si lo sabes conservar sin caer en la monotonía sii es posible...!!'
hombre,'Ánimo compi no tengas meyo jeje pronto te iré a visitarte'
hombre,'No es gusto.. es por amor!'
mujer,'se pasa bien, estamos felices y mas unidos k nunk...'
mujer,'Guapaa!!! Ame tu falda'
mujer,'Ultimo día en mi trabajo :(pero que bonito detalle !!!!!!!'

También se eliminaron frases orientadas a la relación, como aquellas que incluían "*mi novia*", "*mi esposo*" ya que esto podría causar ruido en los resultados de la clasificación posterior.

3.2 Generación de n-gramas

Tomando como punto de referencia el segundo texto del extracto del corpus de mujeres ("*Guapaa!!! Ame tu falda*") podemos generar los bigramas de caracteres *Gu,ua, ap, pa, aa, a!, ¡!(2), ¡_, _A, Am, me, e_, _t, tu, u_, _f, fa, al, ld* y *da*, (tomamos en cuenta símbolos de puntuación). También podemos generar los trigramas *Gua, uap, apa, paa, aa!, a!!, ¡!!, ¡!_, ¡_A, _Am, Ame, me_, e_t, _tu, tu_, u_f, _fa, fal, ald* y *lda*, . La idea de trabajar con gramas es muy simple y tiene la ventaja adicional que puede aplicarse prácticamente para cualquier idioma.

Nuestro modelo hace un análisis estadístico de las apariciones de los gramas en cada una de los textos. Se pretende obtener un conjunto de características definitorias del género de un autor basado en este fundamento léxico de caracteres. Un análisis de este nivel (superficial) no necesita de un procesamiento profundo de los textos, como lo hace un análisis sintáctico (tanto de dependencias como de constituyentes).

El término n-grama refiere a una serie de tokens secuenciales en un documento. La serie puede ser de longitud 1 (unigramas), longitud 2 (bigramas), etc., hasta llegar al n-grama correspondiente. Los tokens usados pueden ser palabras, letras o cualquier otra unidad de información presente en el documento. [15]

El uso de modelos de n-gramas en el PLN es una idea relativamente simple, pero se ha encontrado que es efectiva en muchas aplicaciones. Por ejemplo, modelos del lenguaje a nivel de caracter pueden ser aplicados a cualquier lenguaje, inclusive a otros tipos de secuencias como las del ADN y la música. Otras tareas en donde se ha aplicado esta idea es en la compresión de textos y la minería de datos. [15]

Cada n-grama se convertirá posteriormente en un atributo de tal forma que el algoritmo de aprendizaje que usemos intentará generar conocimiento sobre el uso de los n-gramas por parte de cada autor.

Cada atributo (n-grama) tendrá un valor real asociado que sale de la fórmula:

$$v_i = \frac{frec_{jd}}{Tfrec_j}, \quad (1)$$

donde V_j es el valor del atributo j, $frec_{jd}$ es la frecuencia del atributo j en el documento d y $Tfrec_j$ es la frecuencia del atributo j en la colección de textos.

La generación de estos gramas es automática y a través de la herramientas text2ngram. Un extracto de salida de este módulo es algo como se muestra a continuación (tomando como ejemplo la información mostrada en la sección 3.1):

```
si 4        !! 9
to 3        a_ 3
... ...
```

Esta es solo una muestra de ejemplo de salida, ya que el proceso se hace al analizar todos los textos de un autor (hombre o mujer). El número a la derecha del grama representa el total de apariciones.

A estos archivos se les aplica un proceso de delimitación de fin de línea para que la herramienta text2ngram –que ayudará en la extracción de los n-gramas, pueda generar gramas "falsos" y así poderlos identificar para su eliminación.

Luego, ya estamos en condiciones para aplicar un subproceso parecido al que se llevó a cabo en la construcción del *baseline*, compilando frecuencias de n-gramas.

3.3 Generación de archivo para clasificación

El proceso de clasificación hace uso de un archivo en formato .arff (Weka). Este software contiene una colección de algoritmos de aprendizaje para tareas de minería de datos. La generación del archivo es un proceso semi-automático, que como se ve en la figura 1, necesita un listado de gramas y el archivo que representa al corpus de trabajo. A continuación se muestra un extracto de un ejemplo de archivo generado.

```
@relation generacionweka
@attribute ente numeric
@attribute !!!! numeric
@attribute ento numeric
@attribute ment numeric
... ...
@attribute ades numeric
@attribute genero {'mujer','hombre'}
@data
0,0,0,0,0,1,0,0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,'hombre'
0,0,0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,0,1,0,1,0,0,1,0,0,0,0,1,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,
0,1,0,0,1,'hombre'
... ...
1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,1,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,'mujer'
0,0,0,0,0,0,2,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,'mujer'
```

4. Resultados

Los experimentos fueron desarrollados sobre los datos de un corpus para el problema de identificación de género. Dicho corpus está conformado por textos

obtenidos de redes sociales y blogs. Seleccionamos textos que no sobrepasaran los 300 caracteres y que fueran independientes de las relaciones. Se presentan los resultados de los experimentos para el corpus de 200 y 400 textos. Para la evaluación de los experimentos usamos el 60% de los datos para entrenamiento y el resto para clasificación.

En los resultados que se mostrarán a continuación, usamos el término "profile size" (tamaño del perfil) para representar los primeros n-gramas más frecuentes; por ejemplo, un tamaño del perfil de 40 significa que se usaron solo los primeros 40 n-gramas más frecuentes. Probamos varios umbrales para el perfil y seleccionamos 5 de ellos, como se muestra en todas las tablas de resultados.

Cuando alguna celda de la tabla contiene ND (no disponible) significa que nuestros datos fueron insuficientes para obtener el número correspondiente de n-gramas. Sucede solo con los bigramas, ya que en general hay menos bigramas que trigramas, etc. En estos casos el número total de todos los bigramas es menor que el tamaño del perfil.

Tabla 1. Resultados experimentales para el corpus de 200 oraciones (100 de mujeres y 100 de hombres).

tamaño del perfil	**clasificador**	**tamaño del n-grama**			
		3	**4**	**5**	**6**
40	SVM	48	51	**98**	58
	NB	55	55	**92**	53
	J48	49	50	**98**	50
80	SVM	49	52	**98**	**98**
	NB	55	54	**96**	95
	J48	54	56	97	**98**
120	SVM	52	95	**99**	98
	NB	56	84	**98**	95
	J48	52	97	97	**98**
160	SVM	52	94	**99**	98
	NB	58	82	**98**	95
	J48	53	97	97	**98**
200	SVM	53	95	**99**	98
	NB	56	94	**98**	95
	J48	50	97	97	**98**

La tarea de clasificación consiste en seleccionar características para construir el modelo de espacio de vectores, algoritmos supervisados de entrenamiento y clasificación; es decir, decidir a qué clase pertenece el fragmento de texto –en nuestro modelo de espacio de vectores. En este trabajo presentamos resultados para tres clasificadores: SVM (NormalizedPolyKernel de SMO), Naïve Bayes y J48.

En términos generales vemos los mejores resultados para los gramas más grandes (5 y 6) con una precisión que alcanza hasta un 99%. Esto se obtiene con ambos corpus

de trabajo (200 y 400 textos). Los experimentos se realizaron con un modelo de validación cruzada con 10 iteraciones (10-folds).

Es interesante notar el salto fuerte que hay entre cifras, ya que de pasar de los 50s llega bruscamente a los 80s o 90s, sin detenerse entre 60s y 70s, en ambas pruebas.

Los clasificadores fueron usados motivados por otros trabajos donde han dado resultados aceptables, como en [13].

Tabla 2. Resultados experimentales para el corpus de 400 oraciones (200 de mujeres y 200 de hombres).

tamaño del perfil	clasificador	tamaño del n-grama			
		3	**4**	**5**	**6**
40	SVM	53	**54**	53	49
	NB	**57**	53	52	50
	J48	52	**54**	50	50
80	SVM	55	51	57	**99**
	NB	57	54	56	**97**
	J48	43	54	50	**99**
120	SVM	53	56	**99**	**99**
	NB	56	56	**97**	**97**
	J48	50	55	**99**	**99**
160	SVM	53	98	**99**	98
	NB	55	86	**97**	**97**
	J48	48	98	**99**	**99**
200	SVM	51	97	**99**	**99**
	NB	56	85	**98**	97
	J48	50	98	98	**99**

5. Conclusiones y trabajo futuro

En este artículo se propuso un modelo computacional para la identificación del género de autores de textos cortos. El enfoque se basa en la técnica de n-gramas de caracteres, que entre otras ventajas, se puede aplicar a cualquier lenguaje. Se hicieron pruebas con un corpus de 200 y 400 escritos (sin ningún pre-procesamiento de corrección ortográfica o gramatical). La clasificación se aplicó con los algoritmos NaiveBayes, SVM (SMO) y J48, alcanzando cifras de hasta casi un 100% de clasificación correcta en algunos casos.

Como trabajo futuro se hace necesario probar el modelo con una mayor cantidad de textos cortos y añadiendo características de estilo (riqueza del vocabulario, frecuencia de palabras, por ejemplo) para verificar si esto aumenta los resultados en la precisión.

Referencias

1. Argamon, S., Koppel, M., Pennebaker, J., Schler, J.: Automatically profiling the Author of an Anonymous Text. Communications of the ACM - Inspiring Women in Computing, Vol. 52, No. 2, pp. 119–123 (2009)

2. Bamman, D., Eisenstein, J., Schnoebelen, T.: Gender identity and lexical variation in social media. Journal of Sociolinguistics, Vol. 18, No. 2, pp. 135–160 (2014)
3. Bogdanova, D.: Extraction of High-Level Semantically Rich Features from Natural Language Text. Conference Proceedings II of the 15th East-European Conference on Advances in Databases and Information Systems, pp. 262–271 (2011)
4. Chao-Yue, L. Author Gender Analysis. (retrieved link) (2010)
5. de Iceta, M.: Diferencias cerebrales en función del sexo. Revista web de psicoanálisis, aperturas psicoanalíticas, Vol. 15, (http://www.aperturas.org/) (2003)
6. Kokkos, A., Tzouramanis, T.: A Robust Gender Inference Model for Online Social Networks and its Application to Linkedin & Twitter. First-Monday peer–reviewed journals on the Internet, Vol. 19, No. 9 (2014)
7. Koppel, M., Argamon, S., Shimoni, A.: Automatically Categorizing Written Texts by Author Gender. Literary & Linguistic Computing, Vol. 17, No. 4, pp. 401–412 (2002)
8. Kucukyilmaz, T., Cambazoglu, B., Aykanat, C., Can, F.: Chat Mining for Gender Prediction. Lecture Notes in Computer Science (4243), pp. 274–283 (2006)
9. Muhammad, M., Wolfe, B.: Gender Classification of Mobile Application Reviews (2013)
10. Newman, M., Groom, C., Handelman, L., Pennebaker, J.: Gender differences in language use: an analysis of 14,000 text samples. Discourse Processes, Vol. 45, No. 3, pp. 211–236 (2008)
11. Rosso, P., Rangel, F.: On the identification of emotions and authors' gender in Facebook comments on the basis of their writing style. CEUR Workshop Proceedings 1096, pp. 34–46 (2013)
12. Sarawgi, R., Gajulapalli, K., Choi, Y.: Gender Attribution: Tracing Stylometric Evidence beyond Topic and Genre. Proc. CoNLL '11 Proceedings of the Fifteenth Conference on Computational Natural Language Learning, pp. 78–86 (2011)
13. Sidorov, G., Velásquez, F., Stamatatos, E., Gelbukh, A., Chanona-Hernández, L.: Syntactic Dependency-based N-grams as Classification Features. LNAI 7630, pp. 1–11 (2012)
14. Sidorov, G., Velásquez, F., Stamatatos, E., Gelbukh, A., Chanona-Hernández, L.: Syntactic N-grams as Machine Learning Features for Natural Language Processing. Expert Systems with Applications, Vol. 41, No. 3, pp. 853–860 (2014)
15. Doyle, J., Keselj, V.: Automatic Categorization of Author Gender via N-Gram Analysis. Proceedings of the 6th Symposium on Natural Language Processing, SNLP'2005 (2005)
16. Singh, S., Sarwan, M., Bharshiv, M., Sathe, A.: Gender and Age Classification on the Basis of Blogs.
17. Ugheoke, T., Saskatchewan, R.: Detecting the Gender of a Tweet Sender. M.Sc. Project Report. Department of Computer Science, University of Regina (2014)
18. Yan, X., Yan, L.: Gender classification of Weblog Authors. Proceedings of the AAAI Spring Symposia on Computational Approaches, pp. 228–230 (2006)

Hacia un método para identificación del idioma a través de información acústica

Jesús A. Fortoul-Díaz[1], Ana L. Reyes-Herrera[1], Alejandro A. Torres-García[2], Luis Villaseñor-Pineda[2]

[1] Tecnológico de Monterrey, Puebla, México

[2] Instituto Nacional de Astrofísica Óptica y Electrónica, México

{a01098295, alreyes}@itesm.mx, {alejandro.torres, villasen}@inaoep.mx

Resumen. En la actualidad existen diferentes métodos para la identificación automática del lenguaje hablado basados principalmente en dos enfoques: el primero utiliza información fonética para la tipificación del idioma, y el segundo enfoque utiliza la información suprasegmental extraída directamente de la señal acústica. A pesar de los buenos resultados de los métodos utilizados actualmente para cada uno de los enfoques, en el primero se depende de un estudio lingüístico previo para cada uno de los idiomas a identificar y en el segundo enfoque utiliza un considerable tiempo de cómputo para procesar y obtener la caracterización del ritmo de la señal acústica, haciendo el proceso de identificación del idioma difícil de obtener en tiempo real. En el presente trabajo se presenta un método de identificación del lenguaje hablado, el cual caracteriza la información suprasegmental usando el cálculo de la energía relativa wavelet de cada nivel de descomposición de la señal, así como el cálculo entre segmentos de la señal para capturar el cambio en el tiempo. Los resultados alcanzados son alentadores por lo que se espera continuar el trabajo para llevarlo a una configuración multiclase y con posibilidades de aplicar en tiempo real.

Palabras clave: Identificación automática del idioma, **e**nergía relativa wavelet, información suprasegmental.

Toward a Method for Automatic Language Identification using Acoustic Information

Abstract. Nowadays there are different methods in the field of spoken language identification, mainly based in two approaches: the first uses phonetic information of languages and the second uses suprasegmental characteristic obtained from the acoustic signal. Despite the good results of these methods, the first one depends on a previous linguistic study for each language to identify, and the second one uses a considerable computation time to process and

characterize the rhythm of the acoustic signal, making language identification a little harder to obtain in real time. In this paper, we present a method that characterizes the suprasegmental information calculating the relative wavelet energy of each decomposition level of the acoustic signal, and the calculation between signal segments in order to capture the change in time. The results obtained are encouraging, and it is the reason to continue working with multiclass classification, and apply it in real time.

Keywords: Automatic language identification, relative wavelet energy, suprasegmental information.

1. Introducción

Con el paso de los años, el hombre ha desarrollado técnicas que le permitan simplificar la vida diaria, uno de los puntos en los que se ha centrado principalmente en las últimas décadas es el procesamiento del lenguaje natural. Se ha logrado realizar la identificación de lenguajes, dialectos e idiomas haciendo uso de técnicas que requieren una transcripción fonética, esto quiere decir que la identificación se realiza por medio de similitudes entre sílabas, vocablos o palabras.

Pero qué pasaría si en vez de realizar todo el proceso, se pudiera realizar la identificación por medio de características que no dependan de realizar una transcripción, eso lograría agilizar el proceso, y serviría en casos, en los cuales no se tiene una transcripción fonética. Es por ello que varios investigadores se han dado la tarea de detectar nuevos métodos basados en características suprasegmentales, que se relacionen más con el ritmo, el acento, o la entonación del hablante.

En el presente trabajo se abordará la identificación de distintos idiomas a través de un método que permita su clasificación, tomando como base los proyectos de investigadores que trabajaron con identificación de características suprasegmentales. El método consiste en aplicar la transformada wavelet a las señales de audio, y posteriormente calcular la Energía Relativa Wavelet, con la cual se espera obtener una representación de la señal que proporcione información sobre los niveles en los que la energía es mayor, dependiendo del ritmo del idioma.

La ventaja que ofrece el método planteado en el trabajo es el hecho de requerir menor información para realizar la clasificación de idiomas, en comparación con otros trabajos –descritos en la siguiente sección– lo cual significa que los recursos computacionales a utilizar son menores.

2. Trabajos relacionados

En las últimas décadas se han desarrollado distintos métodos para la identificación de lenguajes, algunos de ellos utilizan la transcripción fonética, es decir se basan en la representación de la voz a través de un alfabeto escrito, realizando la identificación en base a sílabas, palabras o frases similares. Además de ese tipo de métodos existen otros que no requieren el uso de la transcripción y se basan principalmente en características suprasegmentales, es decir, en el ritmo del habla, la entonación, el acento, entre otros.

El trabajo más representativo usando características suprasegmentales es el realizado por Rouas et al. [1], [2]. En este trabajo la identificación de lenguajes se realiza a través del ritmo que posee cada idioma, obteniendo la relación entre los intervalos vocálicos y consonánticos de cada idioma y utilizando los modelos de mezclas Gaussianas (GMM) como técnica de clasificación. Y a través del análisis del ritmo es que logra obtener resultados sobre la identificación entre duplas de idiomas.

Otro trabajo relacionado es [3], el cual nos presentan un método basado en la identificación de idiomas usando características fonéticas y prosódicas, haciendo uso de modelos de mezclas gaussianas (GMM), clasificando los idiomas a través de máquinas de soporte vectorial (SVM) y utilizando un kernel de secuencia probabilística para obtener las características más destacadas de las señales de audio.

Un trabajo interesante es [4], principalmente porque en él se realiza la caracterización de la señal en base a los MFCC de la señal de audio; haciendo uso del algoritmo Fuzzy C-Means logra obtener nuevos vectores de datos, que permiten seleccionar de una mejor forma los atributos más significativos, para posteriormente realizar la identificación en base a modelos de Markov (HMM).

El trabajo presentado en [5] muestra una solución innovadora, basada en la transformada wavelet como método de caracterización; la forma en que realiza la identificación de idiomas es por medio del clasificador Naive Bayes, el cual proporciona muy buenos resultados. Se debe mencionar que en el presente trabajo se estará trabajando con un método similar, teniendo como ventaja una selección de atributos mucho más compacta. La siguiente sección presenta el método propuesto y describe la caracterización propuesta basada en la energía relativa wavelet.

3. Método propuesto

3.1. Transformada wavelet

Para el presente trabajo se hace uso de un método de extracción de características de la señal en el dominio de la frecuencia y al mismo tiempo conserve las características en el dominio del tiempo, este método es la Transformada Wavelet. En un inicio las wavelets eran utilizados en el campo de la geología, en la búsqueda de yacimientos de petróleo, y es ahí donde se observaron las ventajas que ofrecen sobre otros métodos, pues los componentes de la señal procesada siempre mantenían la misma forma, sin importar que la señal sufriera distorsiones como dilatación, compresión o desplazamiento en el tiempo; el único factor que llegaba a influenciar era la wavelet madre que se utilizaba para realizar el análisis [6].

Existen diferentes familias de wavelets madre, entre ellas se encuentra la familia Daubechies, las cuales son wavelets ortonormales aplicadas en análisis de señales discretas; la familia Biortonormales con propiedades usadas en reconstrucción de imágenes. Algunas otras son: la wavelet Haar, la wavelet de Morlet, la familia de wavelet Coiflet, la familia wavelet Symmlet, entre otras. En la siguiente imagen, obtenida de [7], se muestra algunas de las wavelets madre más utilizadas en la actualidad.

El método implementado en el presente trabajo es la Transformada Discreta Wavelet, (DWT) la cual es utilizada principalmente en procesamiento de señales, con

la finalidad de lograr la codificación de la misma señal basándose principalmente en el uso de bancos de filtros para obtener distintos niveles de descomposición [5]. Utilizando un banco de filtros pasa altas se obtienen los coeficientes de detalle, mientras que con el filtro pasa bajas se puede obtener los coeficientes de aproximación.

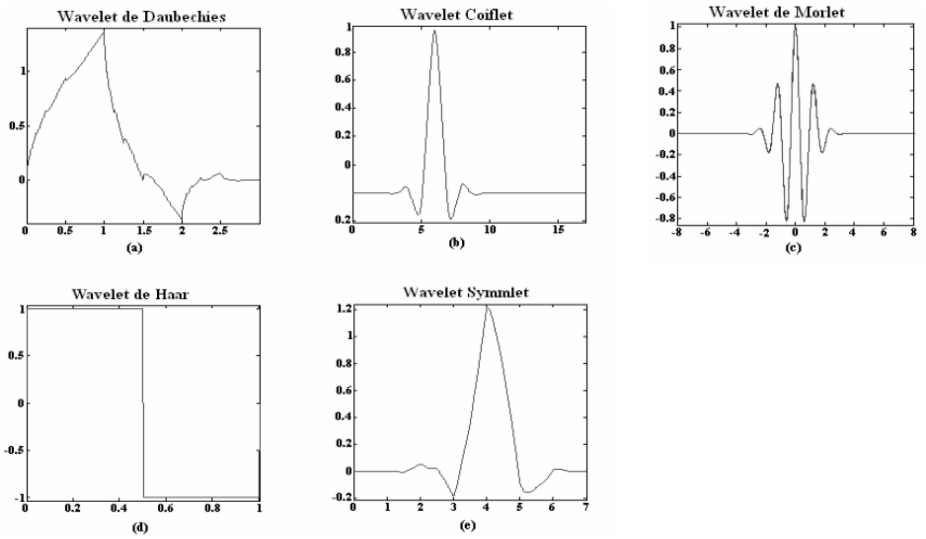

Fig. 1. Muestra representativa de distintas familias de Wavelets: (a) Wavelet Daubechies 2, (b) Wavelet Coiflet 2, (c) Wavelet de Morlet, (d) Wavelet Haar, (e) Wavelet Symmlet 4.

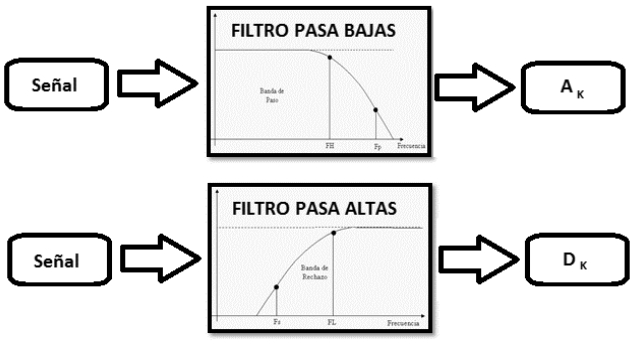

Fig. 2. Proceso requerido para obtener los coeficientes de detalle (Dk) y de aproximación (Ak), a través de un banco de filtros.

La DWT usa un análisis de multi-resolución, en el cual se aplica el algoritmo denominado piramidal, y se puede expresar por medio de los filtros pasa altas y pasa bajas para obtener los coeficientes de detalle y aproximación respectivamente. Debido a que el análisis es del tipo multi-resolución, se obtienen coeficientes de detalle y aproximación en cada una de las diferentes bandas de frecuencia, lo cual significa que cada banda posee la mitad de las muestras de la banda superior [8] (véase la Fig. 2).

Un punto importante a tomar en cuenta durante el análisis de la señal es la forma en que los niveles de descomposición serán obtenidos. Tomando en cuenta el teorema de Nyquist, que expresa que la frecuencia de muestreo mínima necesaria para lograr una grabación de calidad siempre debe ser el doble de la frecuencia de la señal que se pretende grabar [9]:

$$F_s = 2F_0. \tag{1}$$

3.2. Energía relativa wavelet

Para obtener las características más destacadas relacionadas al ritmo del lenguaje se experimentó con los diferentes niveles de descomposición, y el de aproximación, al aplicar el método denominado, Energía Relativa Wavelet (EWR) [8]. La forma de calcular la EWR consiste en obtener la sumatoria del cuadrado de cada uno de los coeficientes para cada nivel "k", ya sea de un nivel de detalle (D_k) o del nivel de aproximación (A_n).

$$E_j = \begin{cases} \sum_k |d_k|^2 & para\ cada\ uno\ de\ los\ niveles\ de\ detalle \\ \sum_k |a_N|^2 & para\ el\ nivel\ de\ aproximacion. \end{cases} \tag{2}$$

Debido a que este método asume que cada nivel (N) posee un porcentaje de la energía total, se procede a calcular la energía total de la señal, la cual consta de los "N" niveles, más la energía en el nivel de aproximación, esto quiere decir que la energía total se conforma de "N+1" componentes:

$$E_{Total} = \sum_{j=1}^{N+1} E_j. \tag{3}$$

Con el cálculo de E_{Total}, se procede a obtener la energía relativa de cada uno de los niveles "j" a través de la siguiente fórmula.

$$EWR_j = \frac{E_j}{E_{Total}}. \tag{4}$$

3.3. Delta de energía

El cálculo de la EWR se realiza a cada determinado tiempo sobre la señal, utilizando un intervalo de tiempo definido (por ejemplo cada segundo), la idea es obtener la energía de cada intervalo y donde el cambio de energía en los estos intervalos proporcione información sobre el ritmo del lenguaje. Para lograr esto, un primer paso es representar cada intervalo con la información más representativa para describir la información suprasegmental. Para ello se calculan los funcionales estadísticos de cada intervalo: la media aritmética (μ), la desviación estándar (σ), el valor máximo, el valor

mínimo, todos ellos respecto a cada nivel de energía; esto quiere decir que la información en cuando a cada nivel se mantiene.

Para capturar la información de la EWR a través del tiempo se calcula el cambio en la energía (delta en la energía), en diferentes puntos del tiempo, la cual representa el cambio de energía de un nivel específico "j", entre dos instantes de tiempo consecutivos:

$$\Delta E1_j = \left| EWR_j\,(t_1) - EWR_j\,(t_2) \right|. \quad (5)$$

De igual forma se puede calcular la delta de energía tomando instantes de tiempo que no sean estrictamente consecutivos, la cual representará el cambio de energía de un nivel específico "j", entre dos instantes de tiempo distantes:

$$\Delta E2_j = \left| EWR_j\,(t_1) - EWR_j\,(t_3) \right|. \quad (6)$$

3.4. Clasificadores

Los atributos seleccionados para realizar la clasificación serán: los estadísticos de cada nivel y los estadísticos de los deltas de energía en el tiempo, con lo cual se espera tener una caracterización descriptiva representativa de cada idioma. En la actualidad existen varias herramientas que permiten llevar a cabo una buena clasificación de datos, para este caso se utilizará la herramienta Weka [10], [11], la cual permitirá realizar la minería de datos que permita encontrar patrones entre diferentes instancias y así determinar las relaciones que describan de una manera conjunta a los datos.

Existen diferentes algoritmos de clasificación en Weka, entre los más comunes se encuentran: J48, Naive Bayes, Random Forest, Máquinas de Soporte Vectorial. A su vez cada uno de ellos puede evaluarse utilizando diferentes opciones de prueba, como lo puede ser: usando el conjunto de entrenamiento, proporcionando un nuevo conjunto, o usar validación cruzada.

El resultado de una buena clasificación se puede observar a través del porcentaje de instancias clasificadas de manera correcta; otra forma en que se puede comprobar el resultado de la clasificación es observando los resultados obtenidos en la matriz de confusión, la cual provee de manera explícita la relación entre falsos positivos y falsos negativos en la clasificación de instancias.

3.5. Procedimiento

Para la realización del proyecto se decidió tomar de referencia los idiomas del corpus OGI que habían sido utilizados en diversas ocasiones dentro del marco referencial. Los idiomas a clasificar son: Alemán, Chino Mandarín, Español, Farsi, Ingles, Japonés, Coreano, Tamil, y Vietnamita.

Por cada uno de los idiomas se usaran 50 audios, con una duración de 10 segundos, en los cuales se escuchan a diferentes interlocutores realizando un relato de su interés. Dos de las características más relevantes de los audios son: la extensión de los archivos es ".wav", y la frecuencia de muestreo es de 8kHz. Es por ello que se decidió trabajar el procesamiento de la señal con Matlab [12], [13], ya que cuenta con una función que permite leer archivos con extensión .wav, posee un toolbox para implementar la

transformada wavelet, y permite conectarse con el api de Weka, para realizar la clasificación necesaria.

Para implementar correctamente la DWT, se debe comenzar por definir los niveles de descomposición que tendrá la señal, es decir, que tan a detalle se analizará. Recordando que la frecuencia de muestreo es de 8kHz, se procede a aplicar el teorema de muestreo, el cual nos dice que la frecuencia de muestreo es el doble de la frecuencia fundamental (de la señal analógica que se digitalizó), por lo tanto al aplicar el teorema se obtendrá que la frecuencia fundamental de los audios es de 4kHz.

Aplicando el mismo teorema para obtener los posteriores niveles de descomposición, en conjunto con el algoritmo piramidal, se llega a obtener la descripción de las frecuencias que se analizan conforme se aumenta de nivel. Para el presente trabajo se decidió realizar una descomposición hasta el nivel 7, con lo cual se logró abarcar frecuencias bajas de hasta 31.25Hz; en la tabla 1 se da una descripción más detallada de los niveles obtenidos.

Tabla 1. Cada nivel de detalle posee límites de frecuencia basados en el teorema de muestreo; el nivel de aproximación abarca desde 0 Hz hasta el límite inferior del último nivel de detalle, en este caso D7.

Nivel de descomposición	Rango de Frecuencias		
D1	2000 Hz	a	4000 Hz
D2	1000 Hz	a	2000 Hz
D3	500 Hz	a	1000 Hz
D4	250 Hz	a	500 Hz
D5	125 Hz	a	250 Hz
D6	62.5 Hz	a	125 Hz
D7	31.25 Hz	a	62.5 Hz
A7	0 Hz	a	31.25 Hz

Una vez que ya se tiene el número de niveles en los cuales se va a descomponer la señal, se procede a realizar la selección de la wavelet madre con la cual se va a realizar el procesamiento de la señal. Llevando a cabo varias pruebas con la familia Daubechies, y con la familia Biortonormal, se llegó a la conclusión de que la mejor wavelet madre que se podría utilizar es la "Daubechies 8", ya que proporciona mejores resultados a la hora de realizar la clasificación.

Automatizando el proceso en Matlab, se logró que el programa accediera a los archivos de audio, los cargara en el espacio de trabajo, y dividiera la señal en periodos de 1 segundo; una vez teniendo los segmentos de la señal se aplicó la transformada wavelet "Daubechies 8", con 7 niveles de descomposición, y con ello se obtuvo un vector de coeficientes por cada nivel. Posteriormente se aplicó el método de la Energía Relativa Wavelet a los vectores, con la finalidad de obtener el porcentaje de energía total que posee cada uno de los niveles; la siguiente figura ilustra de una mejor manera este proceso.

Este proceso se repite para cada uno de los segmentos de la señal, obteniendo como resultado final una matriz en la cual los renglones representan los niveles de descomposición, mientras que las columnas representan cada periodo de la señal

(segundo a segundo). Una vez que ya se tiene la matriz se pueden obtener los estadísticos de los niveles; al mismo tiempo ya se pueden calcular las deltas de energía, ya sea la delta cercana $\Delta E1_j$, o la delta lejana $\Delta E2_j$ para cada nivel.

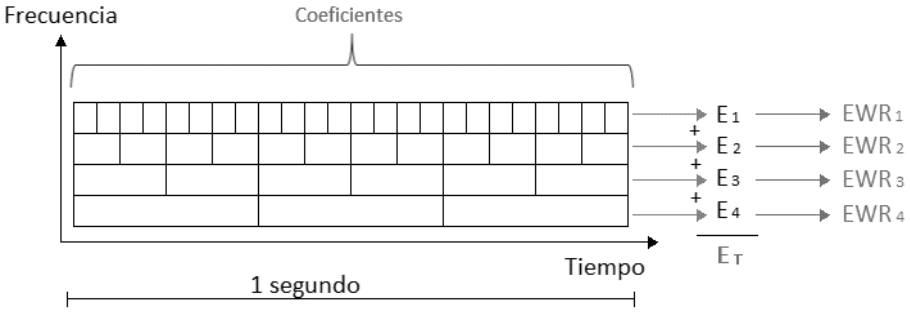

Fig. 3. Representación del proceso que transforma los coeficientes de los diferentes niveles en valores de Energía Wavelet Relativa, tomando en cuenta que la señal analizada es de 1 segundo.

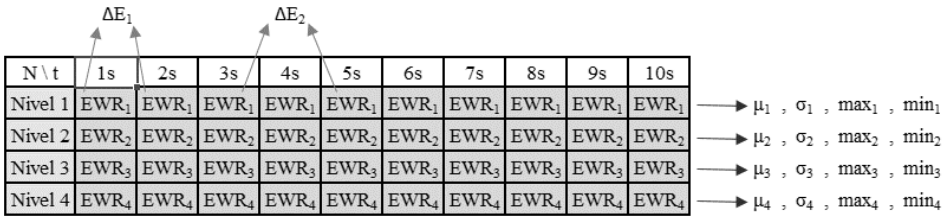

Fig. 4. Representación de una señal de audio de 10 segundos ya procesada; la cual permite obtener datos de la señal como: la media, desviación estándar, máximo, mínimo, y deltas de energía en el tiempo.

Cada una de las deltas de energía generará más información sobre como es el cambio de EWR a lo largo del tiempo; de ellas podemos extraer los mismos estadísticos como se hizo anteriormente en cada nivel. Al igual que en la etapa de lectura de la señal, se programó un algoritmo que permitiera desarrollar un archivo ".arff" de forma automática, usando como atributos los estadísticos de nivel, y los estadísticos de las deltas de energía, y como clase los dos idiomas que se deseaba comparar. Es por ello que cada señal de audio debe contar con el siguiente conjunto de atributos para ser correctamente clasificada.

Fig. 5. Se muestran los 96 atributos distintivos de una señal de audio, los cuales se utilizarán para realizar la clasificación correspondiente.

Una vez que se tiene construido el archivo, se carga al entorno de Weka, donde se realizaron diferentes pruebas con los clasificadores: J48, Random Forest, Naive Bayes, y Maquinas de Soporte Vectorial (SMO), cada uno fue evaluado utilizando validación cruzada con 5 pliegues; con ello se descubrió que los mejores resultados los daba el clasificador SMO, ya que el porcentaje de instancias correctamente clasificada era un poco más alto en comparación a los otros clasificadores.

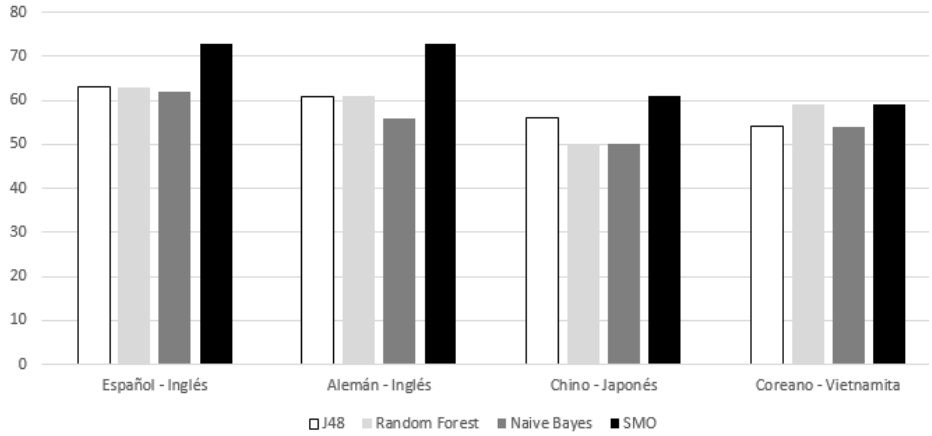

Fig. 6. Resultados obtenidos con diferentes clasificadores, para 4 duplas de idiomas; se observa como en promedio los mejores resultados son obtenidos con el clasificador SMO.

4. Resultados

4.1. Corpus OGI

Para realizar las pruebas, se hizo uso del "OGI Multilanguage Corpus" [14], el cual consta de diversos audios grabados a través de conversaciones realizadas por medio de teléfonos comerciales. Algunos de los idiomas que incluye el corpus son: Inglés, Español, Francés, Alemán, Japonés, Chino Mandarín, Tamil, Coreano, Vietnamita, entre otros; incluyendo un total de 175 llamadas por idioma. Se debe mencionar que la frecuencia de muestreo a la que se grabaron los audios fue a 8KHz, y la grabación fue realizada por medio de una contestadora automática.

4.2. Prueba experimental

Haciendo uso de la wavelet Daubechies 8, y aplicando el método de la EWR, se obtienen los estadísticos necesarios para que por medio del clasificador SMO, se evalúen las clases utilizando 5 pliegues. Con ello se procedió a aplicar el método a todas las duplas de idiomas para obtener el porcentaje de clasificación.

Tomando como referencia la tabla comparativa utilizada en los trabajos descritos en el marco de referencia, [1], [5], se logró determinar que en algunos casos los resultados obtenidos superaban el porcentaje obtenido en otros trabajos.

	Inglés	Alemán	Español	Chino	Vietnamita	Japonés	Coreano	Tamil	Farsi
Inglés	X	73	73	57	53	74	58	78	58
Alemán		X	58	59	53	48	56	62	60
Español			X	61	69	61	48	62	69
Chino				X	49	61	61	71	57
Vietnamita					X	53	59	59	53
Japonés						X	54	65	57
Coreano							X	64	54
Tamil								X	67
Farsi									X

Fig. 7. Resultados obtenidos en la aplicación del método descrito, a las duplas de idiomas.

	Inglés	Alemán	Español	Chino	Vietnamita	Japonés	Coreano	Tamil	Farsi
Inglés	X	73 (59.5)	73 (67.7)	57 (75)	53 (67.7)	74 (67.6)	58 (79.4)	78 (77.4)	58 (76.3)
Alemán		X	58 (59.4)	59 (62.2)	53 (65.7)	48 (65.8)	56 (71.4)	62 (69.7)	60 (71.8)
Español			X	61 (80.6)	69 (62.1)	61 (62.5)	48 (75.9)	62 (65.4)	69 (66.7)
Chino				X	49 (50)	61 (50)	61 (73.5)	71 (74.2)	57 (76.3)
Vietnamita					X	53 (68.6)	59 (56.2)	59 (71.4)	53 (66.7)
Japonés						X	54 (65.7)	65 (59.4)	57 (66.7)
Coreano							X	64 (62.1)	54 (75)
Tamil								X	67 (69.7)
Farsi									X

Fig. 8. Imagen comparativa entre los resultados obtenidos utilizando el método propuesto, y los resultados obtenidos por Rouas et al, que se encuentran ubicados entre paréntesis. Si el resultado obtenido supera al estado del arte, se sombrea en gris.

5. Conclusiones

A lo largo de este trabajo se presentó una técnica innovadora con la cual se pretende realizar la identificación de idiomas, basada principalmente en la obtención de la transformada wavelet de la señal. Complementando a la transformada wavelet, se realizó el cálculo de la energía wavelet relativa, con la que se puede determinar el grado de influencia de cada uno de los niveles de descomposición a la hora de realizar la clasificación.

Observando los resultados obtenidos durante la prueba, se puede comprobar que el método implementado tiene resultados alentadores, ya que en algunos casos los resultados obtenidos en este trabajo lograban superar a los obtenidos por el trabajo de referencia. La ventaja del método propuesto en el presente trabajo respecto a los que ya existen, es que para obtener los resultados de clasificación de instancias correctas solo fueron necesarios pocos datos por idioma (solamente los estadísticos de cada nivel, y los estadísticos de las deltas de tiempo), lo cual abre una puerta para continuar trabajando en él ya que esto permitirá que el procesamiento computacional sea menor, pudiendo dar paso a la realización de un sistema que trabaje en tiempo real.

Como trabajo a futuro se espera implementar diferentes técnicas para lograr una mejor clasificación entre las duplas de los idiomas, con lo cual se espera aumentar el porcentaje de instancias correctamente clasificadas; la primera será realizar un cambio

en como representar la energía de la señal de audio, ya que si se describe mejor se podrá mejorar la exactitud con que se clasifican los datos. Además es necesario incluir un paso de pre procesamiento de la señal para eliminar silencios, ya que eso afecta considerablemente el cálculo de cambio de la energía.

Otro experimento a realizar es en cuanto al cálculo de las deltas de tiempo, las cuales pueden calcularse por medio de una delta más generalizada, pues con ella se englobará más información de la señal, y no solamente representará el cambio entre dos puntos del tiempo, consecutivos o distantes, sino el cambio de energía entre más puntos de tiempo.

Por último se pueden aplicar técnicas de selección de atributos sobre la caracterización actual de la señal de audio, es decir se debe realizar un análisis de los atributos, tal como ganancia de información, mejorar la clasificación, ya que como se sabe, en algunos casos, unos atributos pueden reducir la precisión con la que un modelo puede llegar a clasificar las instancias de forma correcta.

Referencias

1. Rouas, J.-L., Farinas, J., Pellegrino, F., André-Obrecht, R.: Modeling Prosody for Language Identification on Reas and Spontaneous Speech. IEEE ICASSP, Francia, pp. 40–43 (2003)
2. Rouas, J.-L., Farinas, J., Pellegrino, F., André-Obrecht, R. Rhythmic unit extraction and modeling for automatic language identification. Journal Speech Communication, Elsevier, pp. 436–456 (2005)
3. Hosseini Amereei, S.A., Homayounpour, M. M.: Using probabilistic characteristic vector based on both phonetic and prosodic features for language identification. Telecommunications (IST), 5th International Symposium on Tehran: IEEE, pp. 750–754 (2010)
4. Sadanandam, M., Prasad, V.K., Ramana, N., Rao. E.J.: New features using fuzzy c-means alogorithm for automatic language recognition. Computational Intelligence and Computing Research (ICCIC) Coimbatore: IEEE, pp. 1–5 (2014)
5. Reyes-Herrera, A.L.: Un Método para la Identificación Automática del Lenguaje Hablado Basado en Caracteristicas Suprasegmentales (Tesis doctoral). Tonanzintla Puebla: INAOE (2007)
6. National Academy of Science: Wavelets: Ver el Bosque y los Árboles. Recuperado, Beyond Discovery: http://www7.nationalacademies.org/ spanishbeyonddiscovery/mat_008276-03.html (2003)
7. Cortazar-Martinez, O.: Procesamiento Digital de Imágenes Usando Wavelets (Tesis de Ingeniería en Electrónica y Telecomunicaciones). Pachuca de Soto, Hidalgo: Universidad Autónoma del Estado de Hidalgo (2006)
8. Torres, A.: Análisis de Señales Electroencefalográficas para la Clasificación de Habla Imaginada. Revista mexicana de Ingeniería Biomédica, Vol. 34, No. 1, (2013)
9. Morales-Mendoza, L.: Teorema de Muestreo. Procesamiento Digital de Señales, pp. 22–30, DICIS-UG (2011)
10. The University of Waikato: Use Weka in your Java code. Obtenido de https://weka.wikispaces.com/Use+WEKA+in+your+Java+code#Classification-Building a Classifier (2009)
11. Abernethy, M. (s.f.): Data mining with WEKA, Part 2: Classification and clustering. http://www.ibm.com/developerworks/library/os-weka2/ (2015)
12. MathWorks: Multilevel 1-D wavelet decomposition. Obtenido de http://www.mathworks.com/help/wavelet/ref/wavedec.html (2006)

13. Jang, R.: Audio Signal Processing and Recognition. Obtenido de Reading Wave Files http://mirlab.org/jang/books/audiosignalprocessing/matlab4waveRead.asp?title=4-2%20Reading%20Wave%20Files (2005)
14. Cole, R., Muthusamy, Y.: OGI Multilanguage Corpus. Recuperado el 26 de Octubre de 2015, de University of Pennsylvania, Linguistic Data Consortium: https://catalog.ldc.upenn.edu/LDC94S17 (1994)

Similitud de series de tiempo basada en longitud de patrones de la transformada por aproximación móvil

Francisco Javier Garcia-Lopez, Ildar Batyrshin, Alexander Gelbukh

Instituto Politécnico Nacional, Centro de Investigación en Computación, México

b151153@cic.ipn.mx, batyr1@gmail.com, gelbukh@gelbukh.com

Resumen. Se propone medir la similitud de series de tiempo utilizando la longitud de patrones que arroja la Transformada por Aproximación Móvil. Además, se propone un método para la selección automática de ventanas de tiempo y un método para la visualización de los intervalos de tiempo en los cuales las similitudes entre dos series ocurren. El método se aplica a series de tiempo de acciones de empresas obtenidas del sitio de *Google Finance*.

Palabras clave: Series de tiempo, medida de similitud, MAT, selección automática de ventana de tiempo.

Similarity of Time Series based on the Length of the Patterns of the Moving Approximation Transform

Abstract. We propose to measure the similarity of time series using the length of the patterns given by the Moving Approximation Transform. In addition, we propose a method for automatic selection of the time window for this transform and a method for visualization of the time intervals in which the similarities between the two series occur. The method is applied to time series of the stock values of the firms obtained from the Google Finance site.

Keywords: Time series, similarity measures, MAT, automatic selection of time window.

1. Introducción

Una de las tareas que ha tenido más atención en los últimos años en el análisis de bases de datos de series de tiempo es la medición de similitud entre series de tiempo [1]. Varios métodos se han desarrollado en minería de datos de series de tiempo para medir tal similitud [3–5]. En la siguiente sección se abordan algunos conceptos que se han desarrollado para esto.

La Transformada por Aproximación Móvil (MAT por sus siglas en inglés) remplaza los valores de una serie de tiempo por los valores de pendiente de las líneas que aproximan sub-secuencias de la serie [2]. La aproximación se hace por regresión lineal y se utilizan ventanas de tiempo para indicar el número de puntos que se aproximan. La asociación de tendencias locales y distancia de tendencias locales son medidas que se obtienen utilizando la MAT. Estas medidas tienen la propiedad de ser invariantes bajo normalizaciones o transformaciones lineales.

La selección de la ventana de tiempo para la transformada MAT es importante para producir los resultados más significativos. En el presente artículo se seleccionan las ventanas de tiempo buscando que las que brinden mayor información.

Batyrshin et al. [2, 7] formularon el problema de desarrollar medidas de asociaciones negativas entre series de tiempo, esto es, cuando una serie tiene una tendencia a la alza en el mismo periodo que otra serie tiene tendencia a la baja. Por ejemplo, dos empresas competidoras pueden tener dinámicas inversas en la bolsa si el precio de las acciones de una sube al mismo tiempo que el precio de las acciones de la otra baja. En [2] fueron propuestos medidas de asociación de tendencias locales para encontrar asociaciones positivas y negativas entre series de tiempo; además se incluyeron diversos ejemplos de su uso para análisis de asociaciones entre datos financieros, económicos, políticos, etcétera.

Específicamente, en [7] se propuso un método de análisis de asociaciones entre patrones de series de tiempo que tienen asociaciones positivas o negativas basándose en la técnica de tendencias locales. El problema de la búsqueda de estos patrones aparece cuando las asociaciones positivas y negativas entre series de tiempo cambian en el tiempo, por ejemplo cuando los precios de la acción pueden cambiar dependiendo de eventos económicos como el lanzamiento de un nuevo producto, o la alianza de varias empresas, baja de precio de petróleo, etc.

En el presente artículo se propone un método de visualización de patrones de las asociaciones, una medida de similitud entre series de tiempo con diferentes tamaños de patrones. Además se presentan las series en forma de red asociativa y finalmente una forma de selección de ventana de tiempo. En [2] se utilizó la medida coseno para medir similitud después de aplicar la transformación MAT pero en el presente artículo se mide similitud basándose en los patrones continuos en que dos series tienen asociaciones negativas o positivas, utilizando el total de patrones y la suma de sus asociaciones, además se propone un método de visualización de los periodos en que se mantienen las asociaciones.

El resto del artículo se estructura de la siguiente manera. En la sección 2 se hace una revisión de los trabajos relacionados. En la sección 3 se presenta la medida que se propone basada en longitud de patrones. En la sección 4 se describe la metodología para seleccionar automáticamente la ventana de tiempo. En la sección 5 se muestran los resultados obtenidos. Finalmente, la sección 6 concluye el artículo.

2. Trabajos relacionados

En esta sección discutimos los trabajos que abordan el problema de la medición de la similitud entre los series de tiempo. Por las restricciones del espacio, omitimos la

revisión del estado del arte en la selección de las ventanas de tiempo y la visualización de las asociaciones.

Algunos trabajos utilizando distancia euclidiana miden similitud comparando sub-secuencias de las series sin importar que éstas no estén alineadas en tiempo, esto es, que el i-ésimo punto de una serie corresponde con el i-ésimo punto de la otra. Para esto se elige un parámetro que indica qué tanto desplazamiento está permitido, entre mayor sea el parámetro la búsqueda se hace más lenta.

Das et al. [3] proponen un algoritmo aleatorizado basado en programación dinámica para calcular similitud utilizando la sub-secuencia común más larga (LCSS por sus siglas en inglés). El algoritmo toma en cuenta desplazamiento en tiempo, cuya máxima tolerancia está dada por un número entero positivo δ. Además de una transformación lineal que permite comparar series con diferentes valores base (por ejemplo, una serie que varía alrededor del valor 100 y otra que varía alrededor del valor 30) y diferentes escalas. El umbral de tolerancia está dado por un número real ε que toma valores entre cero y uno. Tomando las anteriores consideraciones en cuenta, la medida de similitud se da por l/n, donde l es la longitud de la sub-secuencia común más larga y n es la longitud total de la serie.

Alcock et al. [4] miden similitud basándose en características. Las características que se extraen son clasificadas como de primer y de segundo orden. Las características de primer orden son: media, desviación estándar, asimetría (*skewness* en inglés) y la curtosis. Las características de segundo orden son energía, entropía, correlación inercia y homogeneidad local. Estas últimas características fueron consideradas por su uso en imágenes, por lo que las series de tiempo se transformaron en matrices bidimensionales, para esta transformación primero se hace una cuantización Q de los valores, por ejemplo, si se hace una cuantización de la serie 1, 2, 3, 4 con dos niveles, la serie quedaría de la siguiente forma: 1, 1, 2, 2. El segundo paso es la construcción de la matriz $\mathrm{c}(\mathrm{i},\mathrm{j})$ donde el punto (i,j) representa el número de veces que un número en la serie con nivel i es seguido por un punto con nivel j a una distancia d_1. Los parámetros que mostraron los mejores resultados de acuerdo a los experimentos de los autores fueron $\mathrm{Q} = 3$ y $\mathrm{d}_1 = 1$. Además de las antes mencionadas se usan otras características de segundo orden. Para ello se genera el arreglo unidimensional donde cada valor del arreglo en la posición i es la diferencia entre el valor en i y el valor en $i + d_2$. Este nuevo arreglo tendrá una longitud de máximo $n - d_2$. Donde n es la longitud de la serie y d_2 es un parámetro a variar. De este nuevo arreglo se obtienen las mismas medidas que las de la serie de primer orden.

Lin et al. [5] consideran la distorsión dinámica temporal (DTW por sus siglas en inglés). La DTW además del desplazamiento en tiempo, como el considerado en LCSS, también considera la velocidad, es decir, si una serie cambia más rápido que la otra pero de la misma forma. Para implementar DTW se construye una matriz con las distancias de cada punto de una serie contra cada punto de la serie con la que se compara y se busca, utilizando programación dinámica, el camino en la matriz que minimice su distancia acumulativa, esto es, la distancia óptima que minimice la distorsión. En el cálculo del camino óptimo se suelen establecer restricciones alrededor del camino original que establecen qué tan lejos se permite hacer la búsqueda del camino óptimo para acelerar el cálculo. Las restricciones más comunes son: la banda de Sakoe-Chiba y el paralelogramo de Itakura.

Las referencias de [3–5] se enfocan en medir distancia, sin embargo, hay otra tarea importante en series de tiempo es la reducción de dimensiones. En ocasiones las series de tiempo son tan grandes que aplicando directamente los algoritmos conocidos sería muy costoso computacionalmente. Los métodos más conocidos para reducir dimensiones son: transformada discreta de Fourier (DFT por sus siglas en inglés), descomposición en valores singulares (SVD por sus siglas en inglés) o transformada discreta de ondícula (DWT por sus siglas en inglés: *discrete wavelet transformation*).

Finalmente, Ye at al. [6] utilizan la distancia euclidiana y el coeficiente de correlación de Spearman para comparar mediciones de sensores de vibración de dos diferentes fabricantes. Se combinan ambas mediciones de tal forma que se tienen cuatro clases: las series que tienen coeficiente Spearman alto y distancia pequeña son similares, coeficiente Spearman bajo y distancia grande son disimilares, distancia grande y coeficiente Spearman alto significa que los rangos son parecidos pero diferente escala, por ejemplo, series similares pero con algunos puntos con valores muy alejados entre ellas; distancia pequeña y coeficiente Spearman bajo se considera que tiene ruido que varía en un rango reducido afectando el rango de los puntos de la serie pero no la distancia. Los datos que arrojan los sensores tienen la misma longitud y frecuencia de muestreo.

3. Marco teórico

Una serie de tiempo [7] de longitud n, donde n es un entero positivo, es una secuencia de números reales $x = (x_1, x_2, \dots, x_n)$ correspondientes a puntos en el tiempo $t = (1, 2, \dots, n)$. Una serie de tiempo puede ser denotada simplemente como x. Una ventana de tiempo W_i de longitud $k > 1$ es una secuencia de índices $W = (i, i+1, \dots, i+k-1), i \in \{1, \dots, n-k+1\}$. Se define $x_{W_i} = (x_i, x_{i+1}, \dots, x_{i+k-1})$ a la secuencia de valores de ventana de tiempo correspondientes a la serie x. Una secuencia $J = (W_1, W_2, \dots, W_{n-k+1})$ de todas las ventanas posibles de tamaño k para $1 < k \leq n$ es llamada ventana deslizante.

Una función $f_i = a_i t + b_i$ con parámetros $\{a_i, b_i\}$ que minimice la ecuación

$$Q(f_i, x_{W_i}) = \sum_{j=i}^{i+k-1} (f_i(t_j) - x_j)^2 = \sum_{j=i}^{i+k-1} (a_i t_j + b_i - x_j)^2 \qquad (1)$$

es una aproximación de mínimos cuadrados de x_{W_i}, o regresión lineal. Los valores a_i, b_ise calculan de la siguiente manera:

$$a_i = \frac{\sum_{j=i}^{i+k-1} (t_j - \overline{t_\iota})(x_j - \overline{x_\iota})}{\sum_{j=i}^{i+k-1} (t_j - \overline{t_\iota})^2}, \quad b_i = \overline{x_\iota} - a_i \overline{t_\iota}, \qquad (2)$$

donde:

$$\overline{t_i} = \left(\frac{1}{k}\right) \sum_{j=i}^{i+k-1} t_j \quad \text{y} \quad \overline{x_i} = \left(\frac{1}{k}\right) \sum_{j=i}^{i+k-1} x_j \,. \tag{3}$$

La transformación queda de la forma: $a = a_1, a_2, \ldots, a_m$, donde $m = n - k + 1$. Una propiedad importante de la transformada es que es invariante a transformaciones lineales aplicadas simultáneamente. Esto es,

$$MAT(rx + s, ry + s) = MAT(x, y), \quad r, s \in \mathbb{R}. \tag{4}$$

Se define la asociación entre dos transformadas $a_1 = (a_{1_1}, a_{1_2}, \ldots, a_{1_m})$ y $a_2 = (a_{2_1}, a_{2_2}, \ldots, a_{2_m})$ al producto término por término. En el presente trabajo se consideró únicamente el signo de las pendientes por lo que la asociación será definida de la siguiente manera:

$$A(a_1, a_2) = \left(sgn\left(a_{1_1} \cdot a_{2_1}\right), sgn\left(a_{1_2} \cdot a_{2_2}\right), \ldots, sgn\left(a_{1_m} \cdot a_{2_m}\right)\right). \tag{5}$$

Un patrón positivo (negativo) es una sub-secuencia continua de signos positivos (negativos) con la longitud más larga [7].

4 Selección de la medida

En este artículo, proponemos un método para medir similitud basada en longitud de patrones. La primera aproximación para llegar a esta medida fue considerar un porcentaje de la asociación de patrones que se va mostrar, por ejemplo, el 15% de las asociaciones más grades. El problema de este método es que se excluyen patrones que pueden estar muy cerca del umbral y que podrían ser considerables para medir la similitud entre las series. Por lo anterior se busca medir robustamente la similitud de patrones y seleccionar el tamaño de ventana para que tenga mayor confianza.

Para medir similitud se utiliza la longitud de los patrones entre dos series de tiempo en lugar de la medida coseno que se utilizó en [2] para medir su similitud positiva o negativa. Se considera tanto el número total de patrones como la suma de estos. El principal inconveniente de utilizar la medida coseno es que si el número de pendientes positivas es el mismo que de pendientes negativas es resultado es cero (al considerar las el signo de las pendientes solamente).

Para obtener la lista de patrones entre dos series de tiempo se obtiene el signo de la pendiente, de cada serie y se obtiene la multiplicación entre ellas. Obteniendo así un patrón positivo cuando ambas pendientes son positivas o ambas negativas y un patrón negativo cuando tienen signo diferente. Es decir, positivo cuando hay una asociación positiva y negativo en cuando hay asociación negativa. En la fig. 1 se muestran asociaciones para una ventana de 30 entre dos empresas petroleras.

La fig. 2 muestra la forma propuesta de visualizar las asociaciones utilizando sólo el signo de la pendiente. Los intervalos continuos de valor +1 (–1) forman un patrón positivo (negativo). Por ejemplo, en la fig. 2 tenemos la secuencia de patrones de longitud (10, -3, 57, -2, 10, -2, 97, -3, 15, -2, 22), donde los patrones negativos llevan el signo menos.

El gráfico llega hasta noviembre pues se grafican los puntos iniciales de las pendientes. La medida de similitud por longitud de patrones se aplica a la lista de patrones positivos y negativos. Una vez que se tiene la lista de patrones se obtiene su suma y su longitud y se les considera de acuerdo a la siguiente fórmula:

$$\mathrm{SIM}(x) = \frac{\mathrm{sum}(x)}{k_{max} - k + 1} \cdot \left(\frac{\mathrm{ceil}\left(\frac{k_{max} - k + 1}{2}\right) - \mathrm{len}(x)}{\mathrm{ceil}\left(\frac{k_{max} - k + 1}{2}\right) - 1} \right). \quad (6)$$

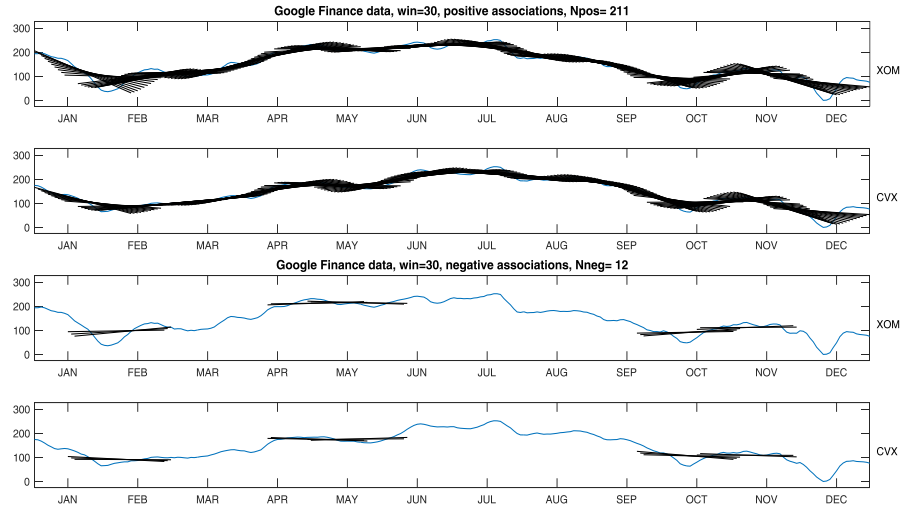

Fig. 1. Asociación entre Chevron y ExxonMobil con ventana de tiempo *k* = 30, datos de 2014.

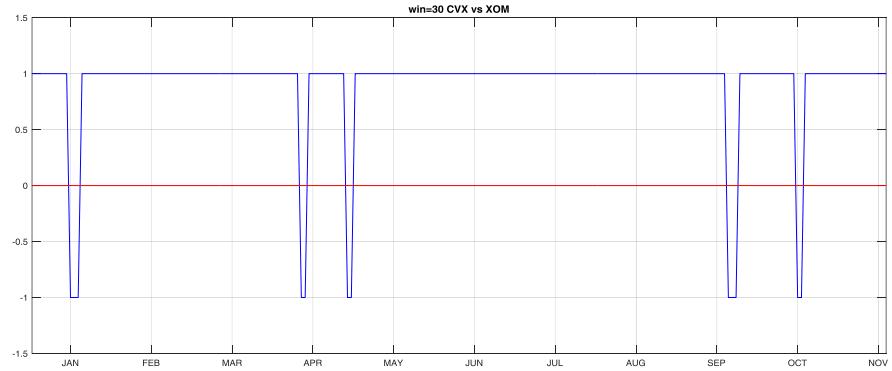

Fig. 2. Los patrones positivos para esta asociación son: {10, 57, 10, 97, 15, 22} mientras que los patrones negativos son: {3, 2, 2, 3, 2}.

En (6) x es la lista de patrones, k es el tamaño de la ventana, k_{max} es el máximo valor que puede tomar la ventana, sum(x) es la suma de todos los valores del patrón,

len(x) es el tamaño de la lista o longitud del patrón y la función ceil asigna el entero positivo más pequeño mayor al número dado.

El primer factor es la suma de los patrones normalizada con respecto a la suma máxima. El segundo factor es la longitud del patrón invertida (afecta negativamente a la similitud) y normalizada. Entre mayor sea la longitud el patrón estará más fragmentado y las series son menos similares ej. El patrón {5} indica mayor asociación que el patrón {1, 1, 1, 1, 1} pues aunque la suma de ambos es la misma en el primero asociaciones están seguidas. Se nota que mientras el tamaño de la ventana se hace más grande, el denominador del primer factor se hace más pequeño esto es porque entre más grande la ventana menor es el número de pendientes que arroja la transformación MAT.

5 Selección de la ventana

Una ventana de tamaño k indica que cada regresión lineal para el cálculo de la MAT será de k puntos, permitiendo un total de $n - k + 1$ ventanas sobre la serie. Se mostró en [2, 7] que ventanas pequeñas detectan los cambios más sensibles, mientras que ventanas más grandes detectan cambios en intervalos de tiempo mayores. Pero la selección de la ventana no se definió con certeza, la cual es una tarea de importancia ya que como se verá, buscando las ventanas apropiadas se pueden obtener una mejor interpretación de las asociaciones.

Como ya se mencionó en la introducción la selección de la ventana de tiempo es importante para saber qué información se está obteniendo. En este trabajo hace la búsqueda de ventanas cuyo valor se ubique entre 2 y la cuarta parte de la mayor ventana posible, lo cual fue determinado experimentalmente, pues con ventanas mayores las series se van haciendo más similares por el hecho de que el número de pendientes que se comparan va disminuyendo. Lo anterior se ejemplifica en la fig. 3.

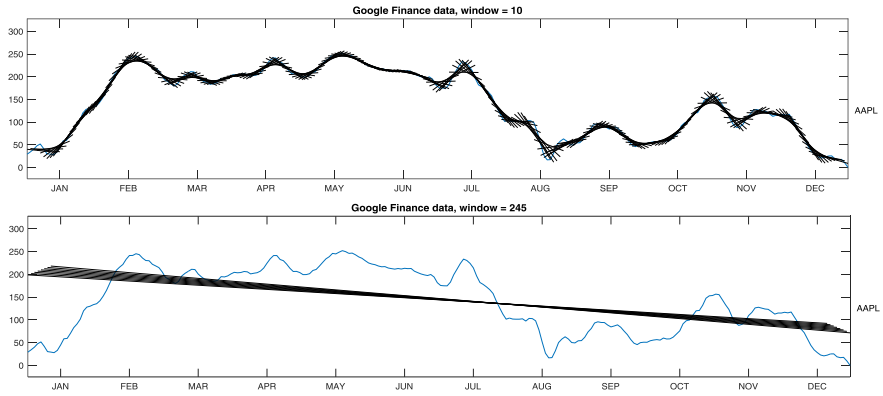

Fig. 3. Comparación del número de pendientes para ventana grande y ventana pequeña.

Se puede afirmar que si dos series están asociadas en una ventana de tiempo pequeña, su similitud es más significativa que cuando el mismo valor de asociación se presenta en ventanas grandes. Las ventanas de tiempo entre más grandes miden sólo como termina una serie con respecto a cómo comienza, es decir si incrementa o disminuye.

El siguiente paso es la selección de la ventana de tiempo. Se propone seleccionarla a partir los valores máximos de los patrones y la longitud de éstos.

Del diagrama como el mostrado en la fig. 4 se obtiene una lista de qué par de empresas es la que tiene el valor más alto para cada valor de ventana (valor máximo por ventana). También se obtiene por cuántas ventanas consecutivas se mantienen ese par como el más alto (longitud del par). Los valores máximos dan información de entre todos los pares, cuáles son los más significativos dado cierto tamaño de ventana, sólo se toman en cuenta los valores mayores a 0.5.

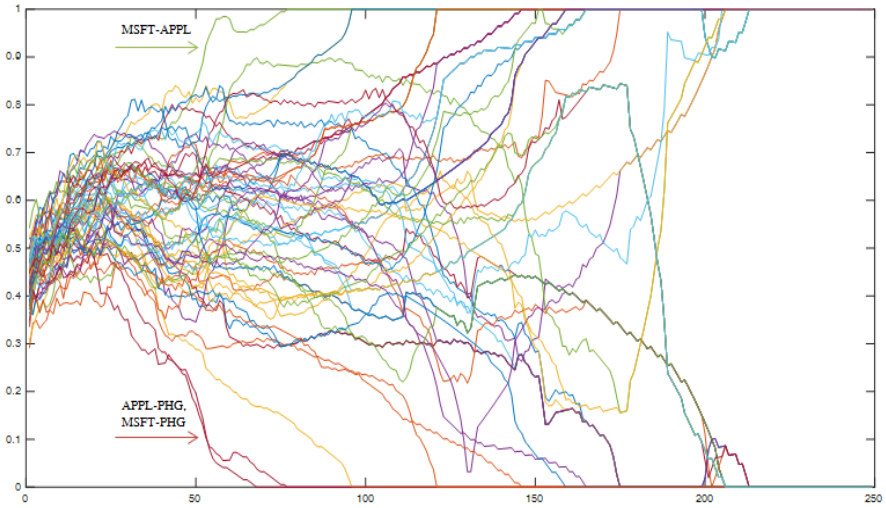

Fig. 4. Similitud positiva utilizando diversos valores de ventana. Empresas de Tecnologías de la información en 2014.

Para sugerir una ventana se le da un peso mayor a la longitud del par que al valor máximo. Una mayor longitud refleja que una similitud dominante de un par de series de tiempo se mantiene para más tamaños de ventana consecutivos. Otra razón para elegir la longitud del patrón es cubrir un mayor intervalo de tiempo. Cabe mencionar que el inconveniente de esta elección es que puede haber otros patrones por debajo del máximo que podrían sugerir otras ventanas además del máximo. Para generar el arreglo de ventanas sugeridas se multiplican los valores de longitud y de valor máximo y el valor mayor de éstos se toma como primera sugerencia para ventana y sucesivamente se hacen las demás sugerencias.

En la fig. 4 se grafica como cambian los valores de similitud de patrones con respecto al tamaño de ventana. Se observa que conforme el valor de la ventana aumenta los patrones se van asociado y quedan ya sea en +1 o 0. Esto se debe a que, como ya se mencionó, los valores mayores de ventana reflejan sólo como empieza y termina una serie contra como empieza y termina la serie con la que se está comparando. Cerca del punto 50 se observa cómo dos series se acercan a cero mientras que una se acerca a 1, esta información es en parte redundante (refleja el hecho de que si dos series son similares entonces si una de ellas es disimilar a una tercera, la otra también lo será) ya que la serie que se acerca a 1 es la de MSFT-APPL, mientras que las que se acercan a

cero son APPL-PHG y MSFT-PHG. Este tamaño de ventana refleja sólo como unas series crecen y otras decrecen gradualmente como se observa en la fig. 5.

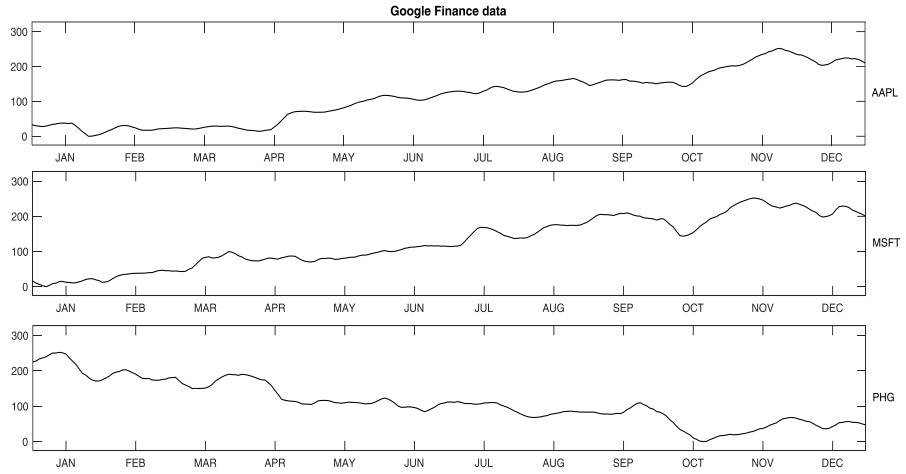

Fig. 5. Datos de 2014.

6 Resultados

En la fig. 6 se muestra el agrupamiento de series de tiempo de ocho empresas petroleras descargadas de *Google Finance*. Para cada par de series se obtienen sus patrones positivos y negativos. Se crea una nueva lista donde se cuenta cuántos patrones de longitud 1, cuántos de longitud 2, hasta la longitud máxima. Se obtiene el 15% de la longitud máxima y todos los valores por debajo de ese porcentaje se eliminan. Algunos pares quedarán sin patrones después de la eliminación pues ninguno de sus patrones fue lo suficientemente largo. Los pares que quedan con patrones son los que se imprimen en el grafo.

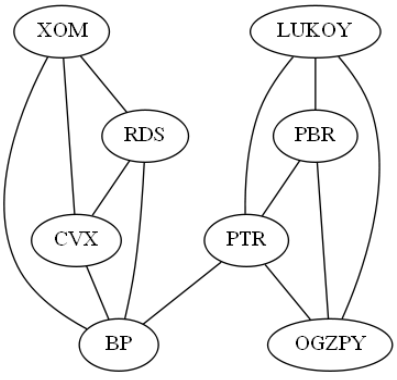

Fig. 6. Agrupamiento de empresas petroleras de países pertenecientes y no pertenecientes a la OCDE. Datos de 2014.

La asociación de patrones de empresas petroleras encontrada que casi completamente separaba en dos clústeres fuertemente conectados, uno con algunos integrantes de las consideradas siete hermanas (cuyos países pertenecen a la OCDE): BP, Chevron, Royal Dutch Shell y ExxonMobil; y otro con empresas petroleras de países no pertenecientes a la OCDE: Gazprom, Petrobras, Lukoil y Petrochina; países que tienen empresas que forman las consideradas nuevas siete hermanas. Los grupos formados están fuertemente conectados y casi completamente separados uno del otro.

Se utilizan datos de Google Finance y separan en datos de 2014 y 2015 como se aprecia en la fig. 7 y fig. 8 respectivamente. Por cada año son aproximadamente 250 puntos. Las empresas de T.I. elegidas por la disponibilidad de sus datos se muestran en la Tabla 1:

Tabla 1. Empresas analizadas en 2014 y en 2015. Elegidas por la disponibilidad de sus datos.

Tecnologías de la información	
NASDAQ:AAPL	Apple Inc.
NASDAQ:AMZN	Amazon.com, Inc.
NYSE:IBM	International Business Machines Corp.
NASDAQ:INTC	Intel Corporation
OTCMKTS:LNVGY	Lenovo Group Ltd.
NASDAQ:MSFT	Microsoft Corp.
OTCMKTS:PCRFY	Panasonic Corp.
NYSE:PHG	Koninklijke Philips NV
NYSE:SNE	Sony Corp.
NASDAQ:FB	Facebook Inc.
NASDAQ:GOOGL	Alphabet Inc.

Petróleo	
NYSE:BP	BP plc
NYSE:CVX	Chevron Corporation
OTCMKTS:OGZPY	Gazprom PAO
NYSE:PTR	PetroChina Company Limited
NYSE:RDS.A	Royal Dutch Shell plc
NYSE:XOM	Exxon Mobil Corp.
NYSE:PBR	Petroleo Brasileiro SA
OTCMKTS:LUKOY	NK LUKOIL PAO

Las series de tiempo que se analizan en el presente trabajo son series financieras, con longitud de 250 puntos aproximadamente y no se requiere reducción de dimensiones. Proponemos una medida de similitud entre series que están alineadas en tiempo, de no ser así la transformada MAT y posterior comparación no se podría llevar a cabo. La alineación obedece a la misma naturaleza de los datos obtenidos de *Google Finance* [8], sería interesante si no se asumiera tal alineación y se aplicara la transformada MAT para encontrar la LCSS, por ejemplo, pero no es el propósito del artículo. El tratamiento que se le da a la serie antes de aplicar la comparación es la transformación MAT.

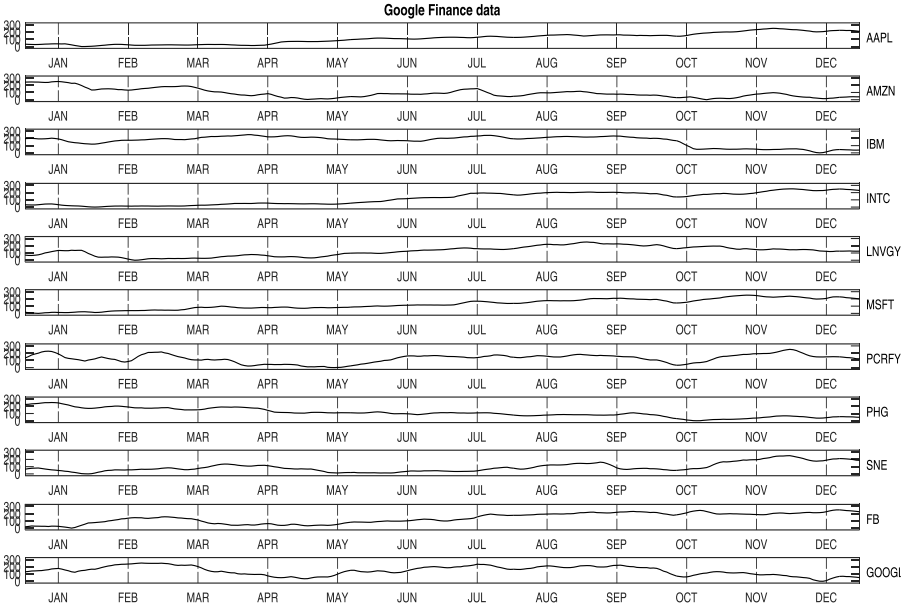

Fig. 7. Datos de TI de 2014

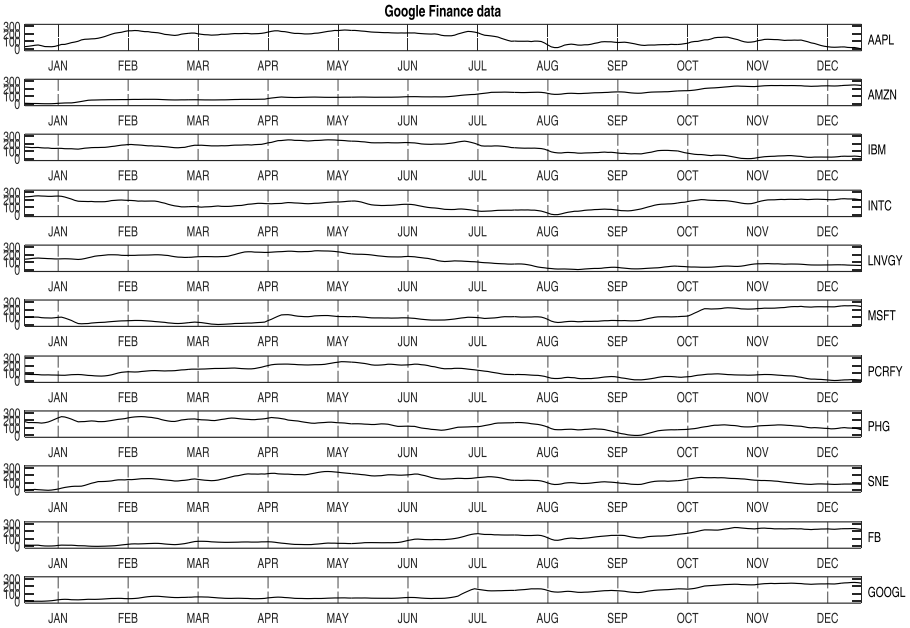

Fig. 8. Datos de TI de 2015

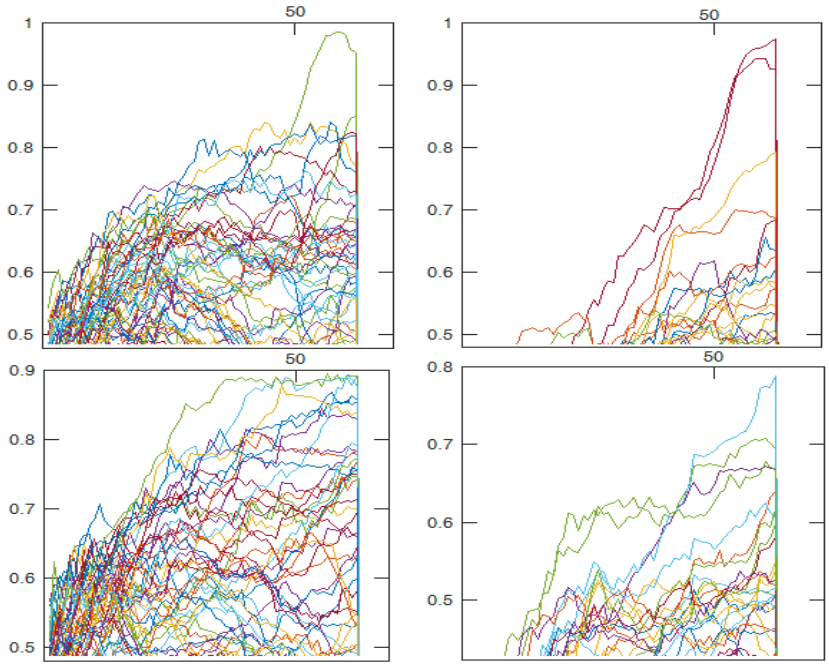

Fig. 9. Funciones de asociación para Empresas de T.I.: Arriba izquierda asociación positiva 2014, Arriba derecha asociación negativa 2014, Abajo izquierda asociación positiva 2015, Abajo derecha asociación negativa 2015

En la fig. 9 se observan las funciones de asociación, el eje *x* es el tamaño de la ventana. A continuación las ventanas sugeridas para cada gráfica. Para 2014 las ventanas sugeridas para valores positivos y las relaciones mayores a 0.8 son:

Ventana = 58: APPL-INT, APPL-MSFT, MSFT-INT, PCRFY-AMZN;
Ventana = 32: PCRFY-AMZN;
Ventana = 44: APPL-INT-LNVGY

Las ventanas sugeridas para valores negativos son de 54, 23 y 62. Pero sólo hay relaciones mayores a 0.8 para la ventana de 62: APPL-PHG, MSFT-PHG.

Para 2015 las ventanas para valores positivos sugeridas son de 39, 46 y 11; mientras que para valores negativos 62, 26 y 64. Las asociaciones positivas se dan entre (mayores a 0.8): APPL-PCRFY, PCRFY-SNE; APPL-PCRFY, PCRFY-SNE, SNE-APPL, SNE-IBM, SNE-LNVGY. Para la ventana de 11 se bajó el offset a 0.65 para mostrar la relación APPL-SNE. No hay asociaciones negativas mayores a 0.8, las más altas son con ventana de 62: IBM-GOOGL e IBM-FB, 0.70 y 0.77 respectivamente.

Una interpretación del porqué de estas relaciones no es sencilla. En el precio de una acción hay un factor de especulación. Además una empresa tiene distintas áreas y en algunas le puede ir bien como en otras mal. Las relaciones por ejemplo entre INT, APPL y MSFT vistas en 2014 se pueden interpretar como una empresa que manufactura procesadores y otras que venden los sistemas operativos. Incluso LNVGY que también

manufactura los equipos aparece en esas relaciones. Sin embargo esta explicación no se sostiene para 2015.

En 2015 existe la asociación entre PCRFY y SNE que además se puede ver desde mediados de 2014. Aunque ambos son competidores en el mercado de los televisores no son los que tienen la mayor parte del mercado (Samsung y LG). A principios de 2015 muchas empresas de televisión incluidas las cuatro mencionadas en este párrafo formaron una alianza (UHDA por sus siglas en inglés) para establecer estándares de calidad de los futuros productos de Ultra-Alta-Definición lo cual puede ser una explicación de la similitud en sus acciones.

Aunque una explicación real debería considerar el ámbito financiero, o el mercado, si sólo se consideran las series de tiempo se observa que lo que arrojan los resultados tiene sentido con respecto su forma visual.

7 Conclusión

En este artículo se propone la medición de similitud entre series de tiempo utilizando la longitud de los patrones y el número de patrones con el mismo signo. Los patrones se obtienen aplicando la transformada MAT y obteniendo la asociación de sus pendientes. A diferencia de utilizar la medida coseno, la similitud de patrones permite medir la similitud de las series de tiempo incluso cuando ésta cambia en tiempo, y además se aprovecha la similitud y disimilitud que mide la MAT.

Los trabajos anteriores [2, 7] usaban ventanas de tiempo elegidas arbitrariamente, o sea, intuitivamente. En este artículo, por primera vez se propone un método para la selección de las ventanas para obtener la información de las series de tiempo con mayor asociación entre distintas ventanas. Los valores de similitud se grafican con respecto al tamaño de la ventana, con lo cual se identifican los pares de series que son más similares para cada ventana. Los máximos de los pares de series, que además se busca que estén separados entre ellos mismos, se consideran como sugerencias para las ventanas de tiempo. Los resultados obtenidos en nuestros experimentos con datos de *Google Finance* tienen una interpretación natural tanto en el caso de las empresas petroleras, como en el caso de las empresas de tecnologías de la información.

También se propone un nuevo método para la visualización de los intervalos de tiempo mucho más clara que la que se ha usado anteriormente [7]. En el método propuesto, las asociaciones se muestran una vez que se ha seleccionado la ventana de tiempo. Este diagrama nos muestra dadas dos series de tiempo si se correlacionan positiva o negativamente en los diferentes puntos en el tiempo. Un ejemplo de la gráfica se muestra en la fig. 2, donde se puede ver que las series de las empresas CVX y XOM están muy correlacionadas en 2014, con la ventana de 30.

Los métodos que proponemos son nuevos y extienden las posibilidades de análisis series de tiempo considerados en [2, 7].

El trabajo a futuro es hacer un análisis más completo incorporando eventos de noticias que podrían ayudar en la explicación del comportamiento de series de tiempo financieras.

Agradecimientos. Este trabajo es parcialmente apoyado por los proyectos SIP 20162204 y 20161958 del Instituto Politécnico Nacional, México, y mediante beca BEIFI del mismo instituto.

Referencias

1. Esling, P., Agon, C.: Time-series data mining. ACM Computing Surveys (CSUR), Vol. 45, No. 1, p. 12 (2012)
2. Batyrshin, I., Herrera-Avelar, R., Sheremetov, L., Panova, A.: Moving approximation transform and local trend associations in time series data bases. Perception-based Data Mining and Decision Making in Economics and Finance, pp. 55–83, Springer (2007)
3. Das, G., Gunopulos, D., Mannila, H.: Time-series similarity problems and well-separated geometric sets. 13th Annual ACM Symposium on Computational Geometry. Association for Computing Machinery (1997)
4. Alcock, R.J., Manolopoulos, Y.: Time-series similarity queries employing a feature-based approach. 7th Hellenic conference on informatics, pp. 27–29 (1999)
5. Lin, J., Khade, R., Li, Y.: Rotation-invariant similarity in time series using bag-of-patterns representation. Journal of Intelligent Information Systems, Vol. 39, No. 2, pp. 287–315 (2012)
6. Ye, J., Xiao, C., Esteves, R.M., Rong, C.: Time Series Similarity Evaluation Based on Spearman's Correlation Coefficients and Distance Measures. Cloud Computing and Big Data, pp. 319–331 Springer (2015)
7. Batyrshin, I., Solovyev, V., Ivanov, V.: Time series shape association measures and local trend association patterns. Neurocomputing, Vol. 175, pp. 924–934 (2016)
8. http://www.google.com/finance

Aplicación web para identificar personalidad, género y edad de usuarios en Twitter

Janet V. Hernández-García[1], Gabriela Ramírez-de-la-Rosa[1], Esaú Villatoro-Tello[1], Héctor Jiménez-Salazar[1], Verónica Reyes-Meza[2]

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

[2] Universidad Autónoma de Tlaxcala, Centro Tlaxcala de Biología de la Conducta, México

2113066463@alumnos.cua.uam.mx, {gramirez,evillatoro,hjimenez}@correo.cua.uam.mx, vrmeza@gmail.com

Resumen. Twitter es una fuente de información para muchas tareas del procesamiento del lenguaje natural. Particularmente para tareas de perfilado de autores, es decir, la tarea de determinar mediante el texto de un autor características demográficas de éste, por ejemplo, género, edad y personalidad. Tradicionalmente este problema se resuelve mediante un enfoque de clasificación supervisado. Uno de los mayores problemas de este enfoque es la necesidad de datos etiquetados (ejemplos) para generar clasificadores confiables; y la complejidad para encontrar datos etiquetados depende del problema. Mientras que es relativamente sencillo obtener información del género, es más complicado obtener información sobre la personalidad de un usuario de Twitter. En este contexto, se presenta una aplicación web que identifica, mediante el análisis del texto de los tuits de un usuario, su perfil que consta de edad, género y rasgos de personalidad. El principal objetivo de la aplicación es que estos usuarios permitan validar la información del perfil generada mediante la respuesta a un cuestionario de personalidad para expandir el conjunto de datos etiquetados. Por otro lado, la aplicación propuesta usa una representación basada en grafos para entrenar modelos para cada problema; esta representación, hasta nuestro conocimiento, no ha sido empleada en clasificación no temática de textos.

Palabras clave: Perfilado de autores, representación basada en grafos, procesamiento del lenguaje natural, aprendizaje supervisado.

Web Application for Analizing Personality, Genre and Age of Twitter Users

Abstract. Twitter is a very useful information source for diverse natural language processing tasks. Particularly, for the author profiling task, *i.e.*,

identifying demographic aspects of an author based on the text that he/she writes. Normally this problem has been solved by means supervised classification techniques. However, one of the biggest problems with this approach is the lack of labeled data (training examples) to build reliable classification models. The complexity of collecting labeled data depends on the problem; for instance, it is easier to find labelled data for genre than data for personality traits. On this regard, we present a web application to identify the profile from a given user through the analysis of his/her tweets. The main goal of the proposed application is that users of our application can validate the results presented to them by answering a personality test, this aiming to expand our labeled corpus. Additionally, our application uses a graph based representation to build the classifiers which, to the best of our knowledge, it has not been used for non thematic clasification problems.

Keywords: Author profiling, graph based representation, natural language procesing, supervised learning.

1. Introducción

Las redes sociales son una de las mayores fuentes de información en Internet a nivel mundial. De acuerdo a Statisticbrain[3], al 1 de diciembre de 2015, el número de usuarios activos de Facebook, la red social con mayor número de usuarios, es de 1374 millones de usuarios; por otra parte Twitter tiene 289 millones de usuarios activos generando un promedio de 58 millones de tuits al día.

Todos estos usuarios de redes sociales generan grandes cantidades de diversos tipos de contenidos como: videos, fotografías, *posts*, revisiones de productos, etc. Utilizando esta información disponible en Internet, se han tratado de resolver problemas de diversos orígenes, por ejemplo: se ha tratado de identificar el estado de ánimo de las personas [10], predecir las fluctuaciones en la bolsa de valores [4], identificar a pedófilos en sitios web de conversaciones [7,21], generar perfiles de usuario [22], entre muchos otros.

En particular, en años recientes, un problema que ha llamado la atención de investigadores en el área de Procesamiento del Lenguaje Natural es el problema denominado *perfilado de autor*, es decir, el problema de identificar, a través del texto que alguien escribe, características demográficas del autor de ese texto; los rasgos más comúnmente identificados son género y rango de edad [11,3,5]. Sin embargo, existen otros aspectos demográficos que son de interés no solo a la comunidad de computación, sino también a áreas de las Ciencias Sociales, particularmente a la Psicología; este aspecto es conocido como la identificación de rasgos de personalidad, mismo que se considera una dimensión más al problema de perfilado de autor.

En este contexto cabe destacar el papel del PAN, quien organiza una serie de eventos científicos y competencias internacionales sobre textos digitales

[3] http://www.statisticbrain.com/social-networking-statistics/

orientados a problemas forenses[4]. En el caso del perfilado de autores, desde el 2013 a la fecha se han enfocado a los problemas de identificación de género y edad en redes sociales. Solamente en el 2015 se contempló la dimensión de rasgos de personalidad. Esto, además de mostrar lo novedoso del problema de identificación de personalidad en redes sociales, habla también de lo difícil que ha resultado generar modelos confiables, pues bajo un esquema supervisado, se necesitan datos etiquetados. Estos datos etiquetados son particularmente difíciles de encontrar para el caso de la identificación de la personalidad de usuarios en redes sociales.

Por lo anterior, en este trabajo se plantea realizar una aplicación web que cumpla dos funciones: *i*) proporcionar un modelo, basado en los datos del PAN-2015, para la identificación de la personalidad, género y edad de usuarios de Twitter, mediante un enfoque de representación basada en grafos; y *ii*) utilizar la plataforma para validar datos mediante un módulo de evaluación, donde los usuarios puedan corroborar o corregir las respuestas de la aplicación web.

Por lo tanto, los objetivos de este proyecto son dos. Primero, evaluar el desempeño de una representación basada en grafos en el problema de perfilado de autor. Hasta nuestro conocimiento, la representación elegida, propuesta por [9] sólo se ha utilizado en tareas de clasificación temática. El segundo objetivo del proyecto es desarrollar una aplicación web gratuita para que usuarios de Twitter puedan conocer su personalidad, y colaborar en la evaluación del modelo generado, mediante un módulo de validación.

El resto del artículo está organizado como sigue: la sección 2 describe algunas plataformas existentes, las cuales se aproximan a los objetivos de nuestro trabajo. En la sección 3 se describe detalladamente la arquitectura del sistema desarrollado. Luego, la sección 4 muestra la funcionalidad de la aplicación web desarrollada. En la sección 5 se evalúa de forma experimental la representación basada en grafos propuesta por [9], utilizada para tareas de perfilado de autor. Finalmente, en la sección 6 se listan las conclusiones del trabajo e ideas de trabajo futuro.

2. Trabajo relacionado

Como se ha mencionado, la identificación de la personalidad mediante el texto es una tarea que tiene retos distintos a los de identificación de género y edad. Por lo tanto, en esta sección se describen de manera muy breve algunos trabajos que han tratado con el problema de la identificación de la personalidad en texto. Posteriormente, se describen las herramientas o aplicaciones más relacionadas con la aplicación desarrollada en este trabajo.

En cuanto a la investigación realizada para la identificación de la personalidad en texto se pueden clasificar trabajos que hacen uso de diccionarios previamente compilados [2,12], como es el caso de LIWC (*Linguistic Inquiry and Word Count*) [20] y la base de datos psicolongüística MRC [6]. Otro grupo de trabajos han

[4] http://pan.webis.de/

dejado a un lado los diccionarios y se han centrado en analizar el contenido de los textos, ya sea el conjunto de palabras independientes o las secuencias de éstas (n-gramas de palabras). Por ejemplo, en [15] los autores evitan el uso de herramientas sociolingüisticas y deciden hacer una representación del texto en forma de n-gramas de palabras de longitud $n = 2$ y $n = 3$.

Otro grupo de investigaciones se han centrado en el análisis de la personalidad en usuarios de redes sociales como Twitter [1,17] y Facebook [16,19]. En general, estos trabajos toman ventaja de la estructura de la red que se forma con otros usuarios así como del comportamiento de cada usuario dentro de esta red para tratar de determinar la personalidad de los usuarios.

Por otro lado, en cuanto a las herramientas y aplicaciones similares a la propuesta en este trabajo, que buscan analizar el texto escrito en redes sociales de usuarios específicos, se pueden listar las siguientes:

AnalyzeWords[5]. Es un sitio experimental que analiza los tuits de una cuenta proporcionada. Se basa en la investigación realizada por Niederhoffer y colaboradores [14], donde la idea principal es que las palabras que se usan para comunicar un tuit revelan no solo pistas de la personalidad, sino del estilo de pensar, estado emocional y la conexión con otras personas. Este sitio permite introducir una cuenta de Twitter, luego realiza un análisis de las 748 palabras más recientes de la cuenta dada. El resultado del análisis es presentado en tres estilos: emocional, social y de pensamiento.

MyPersonality[6]. Fue una aplicación popular de Facebook que permitió a los usuarios realizar pruebas psicométricas reales, y almacenar, con su consentimiento, su perfil psicológico y perfil de Facebook. Actualmente, aunque el sitio ya no está en línea, existe la base de datos recolectada de *myPersonality* que contiene más de 6,000,000 resultados de las pruebas junto con más de 4,000,000 de perfiles individuales de Facebook. Los encuestados provienen de distintos grupos de edad, orígenes y culturas.

Apply Magic Sauce - PredictionAPI[7]. Esta aplicación permite conocer cuál es la personalidad de usuarios en redes sociales. La predicción se basa en los Likes en Facebook. El motor de predicción fue montado por los investigadores del Centro de Psicometría de la Universidad de Cambridge y está basado en el conjunto de datos de *myPersonality* de más de 6 millones de usuarios. Como resultado, se pueden visualizar las predicciones realizadas por el modelo interno; estás predicciones son la edad, genero, personalidad, y la orientación religiosa y política del usuario.

Test de personalidad TP2010. Ésta era una aplicación de Facebook que obtenía información acerca de la personalidad del usuario a través de un test de personalidad, así como también recolectaba datos de las interacciones de usuario con la red social. La meta de TP2010 era descubrir la relación entre los resultados del usuario en la prueba de personalidad y todos aquellos atributos que describen la interacción en Facebook. Esta aplicación también permitía realizar una

[5] http://www.analyzewords.com
[6] http://mypersonality.org/
[7] http://applymagicsauce.com/

comparación de personalidad entre dos usuarios de Facebook. Adicionalmente, podía hacer recomendaciones de amigos basados en compatibilidad, mostrando a las personas, que también habían realizado el test y cuyos resultados eran más similares a los del usuario específico.

En general, las herramientas mencionadas anteriormente proponen distintos métodos para el análisis e identificación de rasgos de personalidad, género y edad (en algunos casos) de información producida por un usuario en redes sociales. La mayoría de estos sistemas incorpora un esquema para recolectar datos; sin embargo, los datos recolectados son en su mayoría en Inglés. A diferencia de la herramienta que proponemos, pues está orientada a recolectar datos en Español. Adicionalmente, los modelos internos con los que las herramientas descritas realizan el perfilado de los usuarios, se basan en su mayoría en los métodos basados en diccionarios o una representación basada en una bolsa de n-gramas, ya sea de palabras o de caracteres. En contraste, en nuestra propuesta proponemos utilizar un método basado en grafos, originalmente descrito y propuesto por [9]. Este método de clasificación utilizando una representación basada en grafos, hasta nuestro conocimiento, no se ha usado en problemas de clasificación no temática de textos.

3. Sistema propuesto

La Figura 1 muestra el esquema general del sistema desarrollado. Dentro de la línea punteada se muestran todos los procesos que se realizan *en línea*, mientras que fuera de la línea punteada se encuentra la generación de los modelos para cada problema, esto es: clasificación de género, edad y cada uno de los rasgos de personalidad del modelo psicológico denominado BigFive [13].

3.1. Construcción del modelo de clasificación

La etapa fuera de línea consiste en la construcción de los modelos de clasificación requeridos. Cabe resaltar que los problemas de clasificación son distintos para cada aspecto demográfico que se trate. Es decir, para el problema de identificación de género se trata con un problema de clasificación binaria, pues las clases sólo son *femenino* o *masculino*. Para el problema de identificación de edad, dado que el conjunto utilizado para generar el modelo consta de cuatro rangos de edad, el problema trata con 4 clases: *18-24* años, *25-34* años, *35-49* años y *50-XX* años. Mientras que para la identificación de la personalidad, se trabajó con 5 clasificadores binarios, uno para cada rasgo del modelo BigFive. De forma que se tienen cinco clasificadores con dos clases: *polaridad negativa* y *polaridad positiva*.

Para la construcción de cada clasificador se implementó el método desarrollado por George Giannakopoulos *et al.* [9] que consiste en dos etapas, la generación de los grafos modelo para cada clase y la segunda etapa que consiste en la construcción del clasificador empleando un enfoque supervisado, donde cada documento es representado por $3 \times N$ atributos (para N igual al número

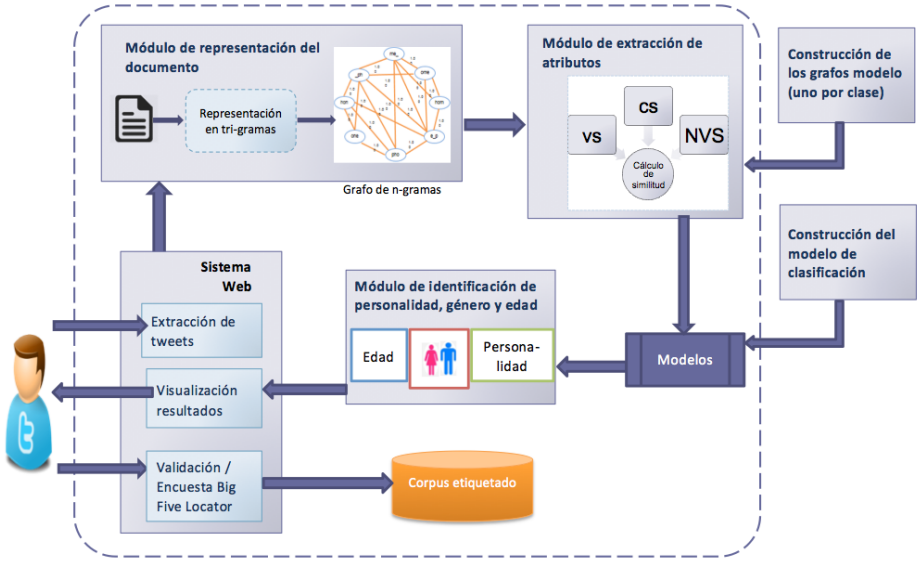

Fig. 1. Esquema general del sistema propuesto para la identificación de personalidad, género y edad de usuarios en Twitter

de clases del problema de clasificación). El cálculo de estos atributos se presenta en la sección 3.3.

Para la primera etapa es necesario crear los grafos modelo de cada clase; para esto es necesario representar cada documento como un grafo, donde cada nodo corresponde a un n-grama dentro del documento, las aristas determinan la co-ocurrencia de dos n-gramas en una misma ventana, y el peso de las aristas representa la frecuencia de co-ocurrencia. Para realizar esta representación se divide el documento en k-ventanas de tamaño t. En este proyecto utilizamos tri-gramas de caracteres con ventanas simétricas de tamaño 4, es decir, cada ventana consiste en el tri-grama analizado, más dos tri-gramas a la izquierda y dos tri-gramas a la derecha.

Una vez que se cuente con los documentos representados mediante un grafo de n-gramas, todos los grafos de la misma clase se combinan en un solo grafo $G^c = \langle V, E, W \rangle$ al que se le denomina grafo de clase c. Esto es, dada una colección de grafos de los documentos de $\mathcal{D}_c$, $G_{d_i} = \langle V_i, E_i, W_i \rangle$, éstos se combinan con $G^c = \langle V \cup V_i, E \cup E_i, W \rangle$, donde W es calculado mediante el valor medio de los pesos de ambos grafos. Al final, el grafo modelo G^c tiene las siguientes propiedades: sus nodos incluyen la unión de los nodos de los grafos individuales, y sus pesos se ajustan de manera que converjan con el valor medio de los respectivos pesos. El grafo resultante captura patrones comunes como de co-ocurrencia y secuencias de caracteres vecinos.

3.2. Representación de documentos

Dentro de la aplicación web, una vez que se introduce el nombre de usuario de Twitter, se proceden a descargar mediante la API de Twitter, los tuits más recientes del usuario en cuestión. Para esto sólo se consideran los tuits originales del usuario. El conjunto de los tuits descargados se conjuntan para formar un solo documento. Este documento de texto se preprocesa para eliminar URLs, remplazar símbolos por una etiqueta S y números por un etiqueta N. Posteriormente, estos documentos se representan en tri-gramas de caracteres.

Una vez que el documento se encuentre representado con el conjunto de tri-gramas de caracteres, se construye el grafo no dirigido $G_i = \{V, E, W\}$, donde V son todos los tri-gramas presentes en el documento, E contiene todas las aristas (Vo, Vd) tal que los tri-gramas de Vo y Vd co-ocurren en una misma ventana. Finalmente, W representa la frecuencia de la co-ocurrencia de dos tri-gramas [9].

3.3. Extracción de atributos

La extracción de atributos consiste en obtener medidas de similitud entre el grafo de un documento d_i con cada uno de los grafos de las clases del problema que se intenta resolver. Sean $G^c = \langle V^c, E^c, W^c \rangle$ el grafo de la clase c y $G_i = \langle V, E, W \rangle$ el grafo del documento d_i. Las medidas de similitud calculadas son las siguientes [9]:

- *Containment Similarity* (CS). Esta medida expresa la porción de aristas de un grafo G_i que son compartidos con un segundo grafo (grafo de la clase c) G^c:

$$CS(G_i, G^c) = \frac{\sum_{e \in E} \mu(e, E^c)}{\text{mín}(|E|, |E^c|)}, \tag{1}$$

  donde $\mu(e, E^c) = 1$ si y sólo si $e \in E^c$, y $|E|$ indica el número de aristas del grafo G_i.
- *Value Similarity* (VS). Medida de similitud que indica el número de aristas del grafo G_i contenidas en el grafo de la clase G^c que tengan el mismo peso:

$$VS(G_i, G^c) = \frac{\sum_{e \in E} \frac{min(w_e, w_e^c)}{max(w_e, w_e^c)}}{max(|E|, |E^c|)}. \tag{2}$$

- *Normalized Value Similarity* (NVS). Es una medida derivada de la anterior donde no se considera el tamaño relativo de los grafos comparados:

$$NVS(G_i, G^c) = \frac{\sum_{e \in E} \frac{min(w_e, w_e^c)}{max(w_e, w_e^c)}}{min(|E|, |E^c|)}. \tag{3}$$

3.4. Identificación de personalidad, género y edad

Como se ha mencionado en secciones anteriores, la aplicación desarrollada trata con tres problemas de clasificación diferentes: clasificación de género, clasificación de edad y clasificación de personalidad (en sus cinco rasgos). En este sentido, se construyeron 7 clasificadores, como se explicó en la sección 3.1. Mediante la utilización de la herramienta Weka [8] se realizaron las clasificaciones según cada problema. El resultado de clasificación es presentada al usuario mediante un módulo de visualización de resultados como se discutirá en la sección 4.

3.5. Módulo de evaluación

Dado que uno de los objetivos de este proyecto es la recolección de datos en español etiquetados con los rasgos de pesonalidad del BigFive, se implementó un módulo de evaluación dentro de la aplicación web. La evaluación consiste en un cuestionario de personalidad: *Big Five Locator* que contiene 25 preguntas con cinco opciones de respuesta. Este módulo consta de tres funciones: *puntaje*, *conversión* e *interpretación*. Finalmente, para la retroalimentación del usuario, el sistema le muestra su personalidad basada, ahora, en el resultado del cuestionario, de modo que le permita contrastar los resultados con el dado por la aplicación web.

Las respuestas al cuestionario se almacenan junto con el conjunto de tuits utilizados por la aplicación web, de forma que con el paso del tiempo y con el uso continuo se puedan almacenar tuits y la correspondiente personalidad, género y edad del usuario. Cabe mencionar que por el momento, no existe retroalimentación entre el *corpus etiquetado* almacenado con la generación de los modelos de clasificación (ver Figura 1); esta etapa es considerada en el trabajo futuro.

4. La aplicación web: Identifying Your Personality-IYP

Con el objetivo de ilustrar el comportamiento de la aplicación web, se realizó una prueba con un usuario de Twitter al que llamaremos *Anónimo* (con cuenta de Twitter *@anonimo*[8]). Una vez que la aplicación descarga los tuits originales del usuario *Anónimo*, y después de realizar todo el procedimiento descrito en la sección anterior, la aplicación propuesta muestra al usuario su perfil, tal como se ilustra en la Figura 2.

En esta Figura 2 puede observarse que el sistema determinó que el género (B en la figura) del usuario Anónimo es *Femenino* y su rango de edad (A en la figura) es de 35 a 45 años. Por otro lado, en cuanto a su personalidad, ésta se muestra en función de cada rasgo del BigFive. Para el usuario *@anonimo*, el

[8] El nombre de la cuenta de usuario es solamente para fines ilustrativos y no corresponde con el usuario de la cuenta https://twitter.com/anonimo

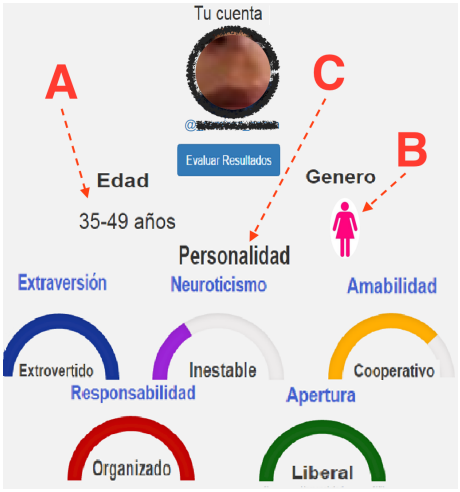

Fig. 2. Ejemplo de la visualización del resultado de edad, género y personalidad para el usuario *@anonimo*

sistema ha identificado que es *Extrovertido*, *Inestable*, *Cooperativo*, *Organizado* y *Liberal*.

La representación de los resultados está basada en la confiabilidad del clasificador para determinar cada uno de los polos (clases) de cada rasgo. Es decir, en el caso del usuario *@anonimo* el sistema arrojó los resultados de clasificación mostrados en la Tabla 1 junto con el grado de confianza del clasificador Naïve Bayes que se utilizó dentro de la aplicación.

Tabla 1. Resultados de la aplicación web para el usuario *@anonimo*. La tercera columna muestra la confianza del clasificador Naïve Bayes de la implementación de Weka que es utilizada para la visualización

Problema	**Clase asignada**	**Confianza (%)**
Género	Femenino	97.3
Edad	34-45	100
	Rasgos de personalidad	
Extroversión	Positivo	100.0
Neuroticismo	Negativo	65.7
Amabilidad	Positivo	80.4
Responsabilidad	Positivo	100.0
Apertura	Positivo	100.0

Algo a notar de la Tabla 1 es que el porcentaje de confianza del clasificador se utiliza para mostrar los resultados de cada rasgo. Observe por ejemplo el

rasgo Neuroticismo en el cual se ha clasificado al usuario Anónimo con la clase negativa en un 65.7 % de confianza, esta información se observa en la Figura 2 en la segunda barra.

5. Validación del modelo de clasificación basado en una representación que usa grafos

En esta sección se presenta el conjunto de experimentos que se realizaron a modo de validar el desempeño de la representación basada en grafos que se describió en secciones anteriores. Note que el modelo que se implementó en este proyecto es el desarrollado por Giannakopoulos y colaboradores [9], mismo que hasta nuestro conocimiento no se ha utilizado en problemas de clasificación no temática.

5.1. Conjunto de datos

Para la evaluación experimental del esquema de clasificación se utilizó un subconjunto del corpus de entrenamiento proporcionado por el PAN 2015 [18] para la tarea de perfilado de autor. El subconjunto que se utilizó contiene tuits en español y cuenta con 100 usuarios diferentes anotados con su género, rango de edad y un valor numérico entre [-0.5, 0.5] para cada rasgo de personalidad. Para nuestras pruebas, cada rasgo de personalidad se consideró como un problema de clasificación binaria, por lo tanto un usuario con valor del rasgo menor o igual a cero se asigna a la clase negativa, mientras que usuarios con valor del rasgo mayores a cero, se asigna a la clase positiva.

5.2. Algoritmos de aprendizaje

Para esta evaluación experimental se utilizaron cuatro algoritmos de aprendizaje de los más representativos y utilizados en el campo del aprendizaje automático. A continuación éstos se describen brevemente:

- *Naïve Bayes (NB)*: Método probabilístico, de los más utilizados por su simplicidad y rapidez, que asume la independencia de los atributos entre las diferentes clases del conjunto de entrenamiento.
- *Árboles de decisión (J48)*: Un algoritmo que permite generar un árbol de decisión, el cual selecciona los atributos más discriminativos basándose en su medida de entropía.
- *Máquina de soporte vectorial (SVM)*: Un discriminante lineal que busca un hiperplano óptimo para separar dos clases.
- Redes neuronales (RBF). Una algoritmo basado en una red neuronal artificial que puede predecir las salidas mediante una combinación lineal de funciones de base radial de los parámetros de entrada.

5.3. Medidas de evaluación

Para evaluar la clasificación se utilizó la medida-F la cual está definida en función de la *precisión* y el *recuerdo* y se define como:

$$medida - F = \frac{(1+\beta^2)Precision * Recuerdo}{\beta^2 Precision + Recuerdo}, \tag{4}$$

donde $\beta = 1$ representa la media armónica entre la precisión y el recuerdo. La función de β es la de controlar la importancia relativa entre las medidas de precisión y recuerdo. Es común asignar un valor de 1 indicando igual importancia a ambas medidas.

5.4. Resultados

Se realizaron tres experimentos: *1)* clasificación de género con dos clases (Femenino y Masculino) , *2)* clasificación de rangos de edad con cuatro clases (18-24, 25-34, 35-49, 50-XX)y *3)* Clasificación de la personalidad para 5 rasgos definido como un problema de clasificación binaria.

Los resultados de las Tablas 2 y 3 son el promedio de la medida-F de realizar los experimentos dos veces sobre dos particiones distintas de 80 %-20 % sobre el conjunto de 100 usuarios.

Tabla 2. Resultados de clasificación (en medida-F) para el problema de Edad y Género

	Género		*Edad*			
	Femenino	Masculino	18-24	25-34	35-49	50-XX
NB	0.81	0.47	0.74	0.57	0.55	0.00
J48	0.81	0.47	0.70	0.47	0.50	**0.60**
SVM	0.81	0.47	**0.83**	0.57	0.55	0.33
RBF	0.78	0.47	0.64	**0.85**	**0.70**	0.00

Tabla 3. Resultados de clasificación (en medida-F) para el problema de Personalidad

	Extroversión		*Neuroticismo*		*Amabilidad*		*Responsabilidad*		*Apertura*	
	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos
NB	0.91	0.97	0.93	0.93	0.92	0.98	0.90	0.98	0.88	0.99
J48	0.87	0.92	0.97	0.96	0.96	0.99	0.99	0.99	0.88	0.99
SVM	0.86	0.91	0.91	0.92	0.91	0.98	0.89	0.97	0.87	0.99
RBF	0.87	0.92	0.94	0.94	0.94	0.98	0.97	0.99	0.87	0.99

Para el experimento *1)* Clasificación por género, puede observarse, en la Tabla 2, que el desempeño de los cuatro clasificadores es similar; sin embargo, siempre es mejor la clasificación para la clase *femenino*.

En la clasificación por edad (experimento *2*) se puede observar que no existe un algoritmo que obtenga el mejor desempeño para todas las clases. Por otro lado, es notorio que la clase que es más fácil de clasificar es la del rango de edad de 18-24 años, mientras que la más difícil es la de 50-xx.

Finalmente, para el experimento *3)* Clasificación de rasgos de personalidad (Tabla 3) se observa que en general todos los clasificadores pueden identificar muy bien todos los rasgos. Es importante notar que pese a los buenos resultados obtenidos en estos experimentos, el corpus con el que se trabajó es muy pequeño por lo tanto no se podría generalizar el uso de esta representación para identificar en más del 90 % el rasgo de personalidad de usuarios de Twitter.

Es necesario realizar un estudio más profundo con una cantidad de datos etiquetados mayor para determinar con mayor confianza la pertinencia de esta representación en la tarea de perfilado de autor tanto para género, edad como para personalidad.

6. Conclusiones y trabajo futuro

En este artículo se presentó una aplicación web *Identifying Your Personality - IYP* para el análisis de la personalidad, género y edad de usuarios de Twitter. El objetivo de esta aplicación es doble; por un lado, se presenta la aplicación como una herramienta para generar corpus etiquetado de usuarios de Twitter con sus rasgos de personalidad de acuerdo al modelo psicológico del Big Five, además de su rango de edad y género. Este objetivo se logró mediante la validación de los usuarios de la aplicación web IYP a través de un cuestionario de 25 preguntas que posteriormente se almacena junto con los tuits descargados al momento de realizar el análisis.

Por otro lado, el segundo objetivo fue evaluar el uso de una representación de documentos basado en grafos (de acuerdo al modelo propuesto en [9]). Este objetivo se cumplió con la construcción de grafos modelos para los tres problemas de clasificación atendidos en la aplicación web. Para la construcción de estos modelos se usaron 100 usuarios del corpus proporcionado por PAN-2015. Los resultados de esta evaluación son prometedores, pero es necesario realizar más experimentos con corpus más grandes que gracias a la herramienta se puede construir.

En este sentido, como trabajo futuro se pretende, por un lado, utilizar la información de usuarios recopilada por la aplicación web para generar mejores modelos que a su vez puedan incorporarse a la aplicación. Por otro lado se realizará una evaluación exhaustiva de esta representación variando los parámetros de construcción de grafos, como el tamaño de las ventanas y el tamaño del n-grama.

Agradecimientos. Agradecemos a la Coordinación de la licenciatura Tecnologías y Sistemas de Información de la Universidad Autónoma Metrópolitana Unidad Cuajimalpa, al CONACyT a través del proyecto No. 258588 y al SNI.

Referencias

1. Adali, S., Golbeck, J.: Predicting personality with social behavior. In: Advances in Social Networks Analysis and Mining (ASONAM), 2012 IEEE/ACM International Conference on. pp. 302–309 (Aug 2012)
2. Argamon, S., Dhawle, S., Koppel, M., Pennebaker, J.W.: Lexical predictors of personality type. In: In Proceedings of the Joint Annual Meeting of the Interface and the Classification Society of North America (2005)
3. Argamon, S., Koppel, M., Fine, J., Shimoni, A.R.: Gender, genre, and writing style in formal written texts. TEXT 23, 321–346 (2003)
4. Bollen, J., Mao, H., Zeng, X.: Twitter mood predicts the stock market. Journal of Computational Science 2(1), 1 – 8 (2011), http://www.sciencedirect.com/science/article/pii/S187775031100007X
5. Burger, J.D., Henderson, J., Kim, G., Zarrella, G.: Discriminating gender on twitter. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 1301–1309. EMNLP '11, Association for Computational Linguistics, Stroudsburg, PA, USA (2011), http://dl.acm.org/citation.cfm?id=2145432.2145568
6. Coltheart, M.: The MRC psycholinguistic database. Quarterly Journal of Experimental Psychology 33A, 497–505 (1981)
7. Escalante, H.J., Villatoro-Tello, E., Juarez, A., Montes-y-Gomez, M., Villaseñor, L.: Sexual predator detection in chats with chained classifiers. In: Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis. pp. 46–54. Association for Computational Linguistics, Atlanta, Georgia (2013)
8. Garner, S.R.: Weka: The waikato environment for knowledge analysis. In: In Proc. of the New Zealand Computer Science Research Students Conference. pp. 57–64 (1995)
9. Giannakopoulos, G., Mavridi, P., Paliouras, G., Papadakis, G., Tserpes, K.: Representation models for text classification: A comparative analysis over three web document types. In: Proceedings of the 2Nd International Conference on Web Intelligence, Mining and Semantics. pp. 13:1–13:12. WIMS '12, ACM, New York, NY, USA (2012), http://doi.acm.org/10.1145/2254129.2254148
10. Golder, S.A., Macy, M.W.: Diurnal and seasonal mood vary with work, sleep, and daylength across diverse cultures. Science 333(6051), 1878–1881 (2011), http://www.sciencemag.org/content/333/6051/1878.abstract
11. Koppel, M., Argamon, S., Shimoni, A.R.: Automatically categorizing written texts by author gender. Literary and Linguistic Computing 17(4), 401–412 (2002), http://llc.oxfordjournals.org/content/17/4/401.abstract
12. Mairesse, F., Walker, M.A., Mehl, M.R., Moore, R.K.: Using linguistic cues for the automatic recognition of personality in conversation and text. Journal of Artificial Intelligence Research JAIR pp. 457–500 (2007)
13. McCrae, R.R.: Cross-cultural research on the five-factor model of personality. Online Readings in Psychology and Culture 4(4) (2002), http://dx.doi.org/10.9707/2307-0919.1038

14. Niederhoffer, K.G., Pennebaker, J.W.: Linguistic style matching in social interaction. Journal of Language and Social Psychology 21(4), 337–360 (2002), http://jls.sagepub.com/content/21/4/337.abstract
15. Oberlander, J., Nowson, S.: Whose thumb is it anyway? classifying author personality from weblog text. In: Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions. pp. 627–634. Association for Computational Linguistics, Sydney, Australia (July 2006), http://www.aclweb.org/anthology/P06-2081
16. Ortigosa, A., Carro, R.M., Quiroga, J.I.: Predicting user personality by mining social interactions in facebook. Journal of Computer System Sciences 80(1), 57–71 (Feb 2014), http://dx.doi.org/10.1016/j.jcss.2013.03.008
17. Quercia, D., Kosinski, M., Stillwell, D., Crowcroft, J.: Our twitter profiles, our selves: Predicting personality with twitter. In: Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on. pp. 180–185 (Oct 2011)
18. Rangel, F., Celli, F., Rosso, P., Potthast, M., Stein, B., Daelemans, W.: Overview of the 3rd Author Profiling Task at PAN 2015. In: Cappellato, L., Ferro, N., Jones, G., San Juan, E. (eds.) CLEF 2015 Evaluation Labs and Workshop – Working Notes Papers, 8-11 September, Toulouse, France. CEUR-WS.org (Sep 2015)
19. Schwartz, H., Eichstaedt, J., Dziurzynski, L., Kern, M., Blanco, E., Kosinski, M., Stillwell, D., Seligman, M., Ungar, L.: Toward personality insights from language exploration in social media. In: Proceedings of the AAAI Spring Symposium Series (2013), https://www.aaai.org/ocs/index.php/SSS/SSS13/paper/view/5764
20. Tausczik, Y.R., Pennebaker, J.W.: The psychological meaning of words: Liwc and computerized text analysis methods. Journal of Language and Social Psychology 29, 24–54 (2010), http://homepage.psy.utexas.edu/homepage/students/Tausczik/Yla/index.html
21. Villatoro-Tello, E., Juarez-Gonzalez, A., Escalante, H.J., Montes-y-Gomez, M., Villaseñor-Pineda, L.: A two-step approach for effective detection of misbehaving users in chats. In: Forner, P., Karlgren, J., Womser-Hacker, C. (eds.) CLEF (Online Working Notes/Labs/Workshop) (2012)
22. Villatoro-Tello, E., Ramírez-de-la-Rosa, G., Sánchez-Sánchez, C., Jiménez-Salazar, H., Luna-Ramírez, W.A., Rodríguez-Lucatero, C.: UAMCLyR at RepLab 2014: Author profiling task. In: Working Notes for CLEF 2014 Conference, Sheffield, UK, September 15-18, 2014. pp. 1547–1558 (2014)

Exploración sobre la construcción automática de un tesauro a partir de un documento

Aarón Ramírez-De-la-Cruz, Héctor Jiménez-Salazar, Esaú Villatoro-Tello, Gabriela Ramírez-De-la-Rosa

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

{aaron.rc24, hgimenezs, villatoroe, a.gaby.rr}@gmail.com

Resumen Es común el problema de requerir el acceso a la información contenida en un texto con un vocabulario más alla del léxico general (por ejemplo, reglamentos o leyes, contratos, e instrucciones de cuidado, entre otros) cuando no se cuenta con diccionarios, glosarios, terminología, o tesauros, todos ellos de un dominio especial. En este trabajo se explora la generación automática de un tesauro a partir del texto de un documento semiestructurado, para establecer un acercamiento a los componentes de este proceso e iniciar un análisis sobre las variables que influyen en la extracción de una parte de la semántica de textos de dominio particular. Se aplicó una adaptación del método SEXTANT a varios textos de dominio especial para generar un tesauro. La revisión de las parejas de términos relacionadas y el texto original nos llevan a concluir una formulación que relaciona las características del texto y productividad del método.

Palabras clave: Tesauro, extracción semántica, estilo de texto.

Explorations on Automatic Thesaurus Construction from a Single Document

Abstract. It is common the need to access to the information at documents which use vocabulary far from the usual lexicon (for example laws, agreements, instructions, etc.) when there is no dictionaries, glossaries or thesauri, all of this fitting for a special domain. In this work it is explored the automatic thesauri generation from the text coming of a semi-structured document. In order to establish a view of the components of this process and begin an analysis about the variables that affect on the extraction of a portion of semantic in texts belonging to a specific domain, it was applied an adaptation of the SEXTANT method to several texts from different domain in order to automatically build, for each one, an specific thesaurus. The review of the related pairs of terms and the source text makes us to conclude the existence of a relation between text characteristics and the productivity of the method.

Keywords: Thesauri, semantic extraction, text style.

1. Introducción

En la actualidad se tiene acceso a grandes cantidades de información textual, mas no contamos con herramientas capaces de aprovechar esta información; en parte, por carecer de recursos lingüísticos suficientes. En efecto, podemos acceder, mediante una búsqueda, a una copiosa masa de documentos, pero en muchos casos es difícil con ellos dar respuesta a una pregunta específica. La navegación en documentos, la búsqueda de respuestas y diferencias sutiles entre documentos, son algunos de los problemas que enfrenta cotidianamente la sociedad del conocimiento y también las motivaciones del presente trabajo.

Una desventaja que tienen los recursos lingüísticos de propósito general es que pueden presentar ambigüedad cuando son aplicados a tareas que tratan con lenguaje o términos de un área específica. Por ejemplo, el término *celular* tiene una acepción si se encuentra en un texto de biología y una diferente cuando se encuentra en un texto de telecomunicaciones. Una forma para desambiguar el significado de los términos es tomar en cuenta su contexto, ya que puede aportar pistas sobre la función de las palabras. Sin embargo ello requiere, a su vez, de recursos lingüísticos para su procesamiento.

Un tesauro es una base de datos léxica que organiza términos de uso común o de un dominio particular. Cada entrada en un tesauro está acompañada por una serie de términos con los que mantiene una o más relaciones; algunas de ellas pueden ser de sinonimia, hiperonimia, hiponimia, entre otras. WordNet [1] es la conocida base de datos léxica en la que las palabras están agrupadas por su significado equivalente en estructuras llamadas *synsets*. Si bien los grupos que constituyen WordNet establecen relaciones semánticas, éstas son generales y no necesariamente coinciden con las usadas en el lenguaje de diferentes áreas de conocimiento.

Normalmente, para construir un tesauro se procede partiendo de un corpus grande de especialidad, y con él se genera el tesauro para después procesar textos de un dominio. Por esta razón hay diversos y eficientes algoritmos que automáticamente generan un tesauro [2], los cuales utilizan grandes cantidades de texto para ser aplicados. Por otro lado, es común el problema de requerir el acceso a la información contenida en un texto con un vocabulario más alla del léxico general (por ejemplo, reglamentos o leyes, contratos, e instrucciones de cuidado, entre otros) cuando no se cuenta con diccionarios, glosarios, terminología o tesauros, todos ellos de un dominio especial. Nuestro objetivo es conocer el alcance de la extracción semántica sobre un documento mediante el análisis de los parámetros que intervienen en él, con la finalidad de asegurar un nivel de precisión sobre el recurso generado.

En este trabajo se explora la generación automática de un tesauro a partir del texto de un documento semiestructurado, para establecer un acercamiento a los componentes de este proceso e iniciar un análisis sobre las variables que influyen en la extracción de una parte de la semántica de textos de dominio particular. Específicamente, se programó un método de extracción de pares de términos relacionados. El método está basado en conocimiento sintáctico [3], lo cual permite mejorar la selección de los contextos de los términos cuando no se

cuenta con un texto grande. Tanto la selección de los textos fuente para construir el tesauro como las evaluaciones de las parejas fueron actividades apoyadas por expertos en dominios específicos. Una vez que se obtuvieron los resultados se llevó a cabo un análisis para conocer la utilidad de este enfoque.

El resto del documento está organizado de la siguiente forma. En la sec. 2 se describen las características principales de trabajos orientados a la construcción automática de tesauros. La sec. 3 se dedica a exponer la adaptación hecha para el español del método SEXTANT. La descripción de las condiciones creadas para la aplicación del método adaptado y los resultados obtenidos se presentan en la sec. 4. Finalmente se lleva a cabo un análisis de los factores que influyen en la extracción de los términos relacionados a partir de un texto.

2. Trabajo relacionado

Los métodos más usados para generar un tesauro reúnen grandes colecciones de texto; generalmente se trata de tesauros de uso, como el caso de [2] o [4] que se apoyan en el British National Corpus (100M de palabras), los cuales son métodos impracticables en nuestro caso debido a que los documentos considerados en este trabajo no pasan de 10 mil palabras. Brevemente, esta sección se refiere a métodos que pueden verse complementarios al que aquí se aplicó.

En [5] proponen la construcción automática de tesauros basada en conceptos formales. La motivación de utilizar este enfoque es debido a que las representaciones de texto tradicionales basadas en un modelo vectorial ignoran las relaciones conceptuales entre términos, como pueden ser hiperonimia e hiponimia. Las pruebas de este trabajo fueron realizadas con el texto de un documento [6] con temática de expansión y mejora de consultas en sistemas de recuperación de información.

El análisis de conceptos formales es utilizado para la extracción de relaciones entre términos dentro de un contexto. Un concepto formal es definido como una tupla de tres elementos, (G, M, I), donde G son objetos, M son atributos y I es la relación binaria entre G y M. Para establecer relaciones jerárquicas entre conceptos utilizan una retícula, en la que cada nodo representa un concepto y las aristas son las relaciones de superconcepto y subconcepto. De esta forma, la relación superconcepto y subconcepto entre conceptos formales juegan roles de hiperónimo-hipónimo, y los que se encuentran un nivel abajo del mismo superconcepto pueden considerarse cohipónimos. La generación del tesauro consiste en una retícula de conceptos relacionados entre sí.

El trabajo de [7] está enfocado en la construcción de una ontología de términos legales. El corpora utilizado son 57 códigos pertenecientes a leyes francesas. Mediante un analizador sintáctico se identificaron 500 000 términos, que incluyen sustantivos, verbos, adverbios, adjetivos y las dependencias sintácticas entre éstas categorías gramaticales (e.g. sujeto de un verbo, objeto de un verbo, adjetivo de un sustantivo). Para identificar las relaciones entre los términos se

utiliza la información mutua (IM) entre palabras, usando la siguiente fórmula:

$$IM_{cw} = log\left(\frac{f_{cw}}{f_c f_w} + 1\right), \tag{1}$$

donde c son las palabras del contexto del término; w son los términos base; f_{cw} es la probabilidad conjunta de c y w en el corpus; f_c y f_w las frecuencias individuales de c y w, respectivamente. El resultado final es una lista de 103 994 términos, cada uno relacionado al menos a otro término.

Como se ha dicho, en este trabajo se eligió el método SEXTANT [3], el cual consideramos viable por no exigir un corpus grande; en cambio, el método se apoya en el análisis sintáctico para extraer contextos de los términos y encontrar parejas relacionadas. Dicho método será presentado en la siguiente sección a través de una adaptación al español.

3. Descripción del método

El método SEXTANT (Semantic Extraction from Text Via Analyzed Network of Terms, Extracción semántica de texto mediante análisis de términos relacionados) [3] emplea el contexto de las palabras para descubrir similitudes entre ellas. Se basa en la hipótesis de que las palabras que son usadas en un contexto similar a lo largo de un corpus de texto están relacionadas semánticamente. Dicho método está presentado en el libro de Grefenstette [3] apoyándose en reglas de la sintaxis inglesa.

El texto utilizado como corpus es dividido por oraciones que terminan con un punto (.). Dado que este signo no es exclusivo del final de una oración, se utilizó una lista de abreviaturas que no son consideradas como fin de línea (e.g., i.e., etc., Art., Vol., Dr., etc.). Las cifras numéricas con decimales (e.g. 3.14159) son ignoradas. Este conjunto de abreviaturas puede ser complementado con más elementos. Posteriormente cada oración es etiquetada con las partes del discurso (Part-Of-Speech) para conocer la categoría gramatical de las palabras (sustantivos, verbos, adjetivos, adverbios, etc.), utilizando el etiquetador Tree-Tagger [8]. En la siguiente descripción se presenta la adaptación hecha con reglas gramaticales del español.

El procedimiento de SEXTANT se realiza en cinco etapas secuenciales. La unidad que utiliza son oraciones simples, formadas por un sujeto, un verbo y el complemento.

Etapa 1: verbo principal. Se parte del hecho que una oración simple posee sólo un verbo conjugado [9]. Tomando en cuenta que en el proceso de etiquetado de texto puede haber casos de asignación incorrecta de categorías, se previeron cuatro casos para detectar dichos errores:

1. una conjunción o palabra clítica antes del verbo: *que le presenten, tuvieron que caminar*;
2. el verbo está en infinitivo, gerundio o pasado participio y está después de un sustantivo: *decisión tomada*;

3. antes del verbo aparece un artículo: *la solicitud, el cantar de las aves*;
4. una negación antes del verbo: *no indica, ningún proceder.*

Si una palabra fue etiquetada como verbo y cae en uno de los casos anteriores, se descarta como verbo principal. También mediante este procedimiento las oraciones compuestas (coordinadas o subordinadas) son segmentadas y conservadas como oraciones simples.

Etapa 2: sintagmas nominales. La estructura utilizada para capturar los sintagmas nominales es *Determinante + Núcleo + Complemento de la oración.* En la Tabla 1 se muestran las categorías gramaticales para cada componente de la estructura anterior. El núcleo puede ser secuencias con más de un sustantivo, como en el caso de las entidades nombradas.

Tabla 1. Categorías gramaticales permitidas por componente

Componente	Tipo	Ejemplo
Determinante	Artículo, demostrativo, posesivo, numeral, interrogativo, exclamativo.	los, esa, mi, tres, cada, cuántos, ¡qué!
Núcleo	Sustantivo propio, sustantivo común.	entidad, Universidad
Complemento	Complemento del sustantivo	fue velozmente

Etapa 3: sujeto principal. Utilizando los sintagmas detectados en la Etapa 2, el sujeto será el núcleo (un sustantivo o más). Si existen adjetivos después del núcleo, éstos se agregan al sujeto debido a que pueden formar parte de él.

Etapa 4: contexto del sujeto principal. Si del sintagma original de la Etapa 2 se omite el verbo principal (Etapa 1) y el sujeto principal (Etapa 3), el contenido restante es el contexto del sujeto principal. Este conjunto puede incluir sustantivos (propios y comunes), verbos y secuencias de sustantivos con adjetivos.

Etapa 5: cálculo de similitud contextual. La similitud entre dos sujetos S_i y S_j es calculada utilizando el contexto c_i y c_j, respectivamente. Se utiliza el coeficiente Jaccard, definido como:

$$sim(S_i, S_j) = sim(c_i, c_j) = \frac{c_i \cap c_j}{c_i \cup c_j}. \tag{2}$$

Una vez que se calcularon las similitudes de todos los sujetos, se ordenan de manera descendente y se presentan en el formato $[sim(S_i, S_j)]$ $[S_i]$ $[S_j]$
Este procedimiento es del orden $O(n^2)$, donde n es el número de sujetos en el corpus utilizado.

Finalmente para construir el archivo de tesauro, se toman los primeros pares de cada sujeto. Por defecto se eligieron 10, aunque éste puede cambiarse.

4. Experimento

El experimento descrito en esta sección está orientado a identificar las variables que influyen en la cantidad de parejas de términos relacionados. Para ello se tomó de cada uno de los textos utilizados tres porcentajes para generar términos relacionados. Una vez validadas las parejas de términos se analizaron las posibles variables que influyen en la producción del método.

Se describen a continuación los datos utilizados, la aplicación del método presentado en la sección anterior, asimismo, los resultados obtenidos.

4.1. Características de los documentos

Los documentos pertenecen a tres dominios diferentes: pedagogía, procesamiento del lenguaje natural, y psicología computacional. Dos de los cuatro autores proporcionaron su documento en formato de texto plano; los otros dos autores entregaron su documento como archivo PDF de los cuales se extrajo el texto y se almacenó como texto plano. La Tabla 2 muestra el tamaño de cada texto empleado para realizar el experimento.

Tabla 2. Características de los documentos

Texto	Número de oraciones	Número de palabras	Promedio de palabras por oración
1	159	5069	32.68
2	205	6124	30.68
3	197	5189	27.32
4	266	10014	38.64

4.2. Descripción del experimento

Previo a aplicar el método SEXTANT para construir el tesauro, de cada archivo de texto se eligieron aleatoriamente el número equivalente al 50 % y 75 % de las oraciones totales. Se generaron tres tesauros, uno para cada porcentaje de texto utilizado de los documentos (incluyendo el 100 % de las oraciones).

Para efectos de esta sección, se establecen las siguientes definiciones:

- **pareja:** relación semántica establecida entre los sujetos (S_i, S_k), en donde S_i y S_k $(k \neq i)$ son entradas del tesauro;
- **pareja útil:** la relación semántica (S_i, S_k) es válida;
- **grupo:** conformado por 10 parejas, de la forma *[(S_i,S_1), (S_i,S_2), (S_i,S_3), ..., (S_i,S_{10})]*

Para este experimento, los elementos del tesauro se ordenaron de forma decreciente con base en la suma de la similitud entre las parejas de cada grupo, por lo que las primeras entradas del tesauro son aquellas cuya similitud acumulada es mayor. De cada tesauro ordenado se extrajeron nueve grupos de 10 parejas cada uno y se presentaron a los respectivos autores para que indicaran la utilidad de las parejas.

Se entregó a los autores tres archivos de texto (uno para cada porcentaje de texto utilizado), cada uno con nueve grupos, correspondientes al tesauro generado a partir del documento que proporcionaron. Se muestra en la Fig. 1 el formato utilizado para que los autores marcaran las parejas que constituyen una relación semántica válida. En dicho formato de ejemplo, `término 1` y `término 3` son relaciones válidas con `entrada`.

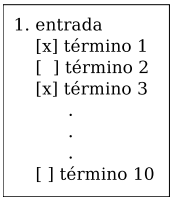

Fig. 1. Formato de un grupo de parejas

4.3. Resultados obtenidos

En la gráfica de la Fig. 2 aparece en el eje horizontal el porcentaje de texto utilizado y en el eje vertical, el número de parejas válidas identificadas por los autores.

5. Análisis de resultados

Los resultados resumidos en la Fig. 2, presentan dos textos que tienen líneas crecientes, uno línea decreciente, y otro no monótona. En el último caso, se detectaron problemas con el preprocesamiento, aún así se consideró útil su análisis. El texto que se comporta decreciente consideramos que pudo influir su sesgo multitématico. De esta manera, no en todos los casos las parejas aumentan al incrementar el tamaño de texto, pero sí hay una tendencia de aumento; lo cual, como veremos se relaciona con otros factores.

Por otro lado, la revisión de las parejas que produjo el método fue hecha tomando muestras de tres niveles de similitud: alta, media y baja. Dichas muestras se realizaron en cada uno de los tamaños de los textos de prueba: 50 %, 75 % y 100 %. En la Tabla 3 se muestra, para cada texto, el número de parejas válidas en cada nivel de similitud, además, la mínima similitud de las parejas que fueron

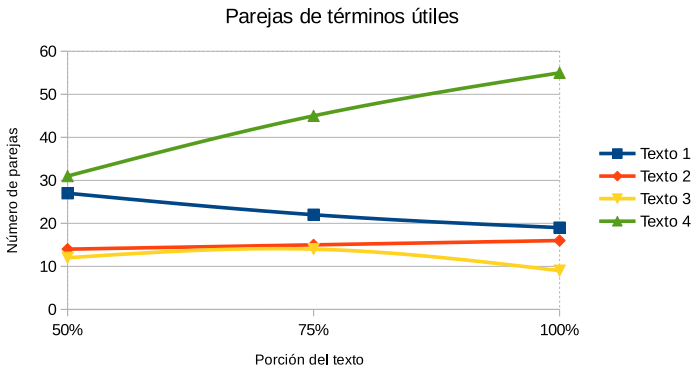

Fig. 2. Parejas útiles

elegidas por los revisores. Lo que se puede observar en esta tabla es que al variar el tamaño de texto (50 %, 75 % y 100 %) el mayor número de parejas válidas se concentra en el nivel alto de similitud. Esto permitiría establecer un umbral para elegir parejas válidas; en el caso de los textos usados en el experimento la similitud mínima promedio de las parejas elegidas en el nivel alto es 0.72.

Tabla 3. Similitud mínima por nivel

Nivel	Texto 1	Texto 2	Texto 3	Texto 4
Alto	28 (0.55)	21 (0.84)	18 (0.53)	51 (0.98)
Medio	22 (0.15)	14 (0.53)	11 (0.42)	40 (0.13)
Bajo	18 (0.03)	10 (0.08)	6 (0.11)	40 (0.05)

En lo que sigue hacemos un análisis sobre el léxico de los textos con el fin de extraer elementos que permitan caracterizar la productividad del método. Como se ha explicado, el método construye las parejas sacando provecho de la sintaxis del texto. En particular, el uso de la puntuación tiene influencia en la definición de contextos puesto que delimita sintagmas. En los textos utilizados se pudieron constatar diferencias sobre el uso de la puntuación. Bien que este elemento sintáctico sea parte del estilo del autor, el resultado conduce a que efectivamente deben adecuarse las reglas sintácticas utilizadas para identificar sintagmas en las cuales la puntuación no sea decisiva.

Asimismo, puesto que el método se basa en la similitud léxica de contextos, la cohesión del texto influye en los valores de similitud entre pares de palabras y en la selección de éstas para ser parte del tesauro. Por ejemplo, como el uso de pronombres constituye un elemento de cohesión, se espera que los textos que utilicen más pronombres tengan menos parejas y viceversa. En un conteo del

uso de pronombres en cada texto se observó que aquél con mayor proporción de pronombres tiene mayor cantidad de parejas. Es decir, las diferencias observadas en la obtención de parejas útiles se debe a otros factores (por ejemplo, la repetición de palabras). Por último se calcularon algunas medidas para precisar la influencia del estilo sobre los resultados del método. Se determinó para cada texto T la cantidad de términos con una sola ocurrencia, $H(T) = \{x | frec(x, T) = 1\}$; la ponderación de términos de ocurrencia unitaria por tamaño de vocabulario, $PH(T) = |H(T)|/|V(T)|$, donde $V(T)$ es el vocabulario de T; la suma del número de oraciones que utiliza cada uno de los términos, $R(T)$; y la proporción de $R(T)$ con respecto al número de oraciones, excluyendo palabras cerradas, $Rec(T)$. Estas medidas se presentan en la siguiente tabla, al igual que el número de parejas producidas por el método para cada texto.

Tabla 4. Medidas de los textos utilizados

T	$\|V(T)\|$	$\|H(T)\|$	$PH(T)$	$R(T)$	$Rec(T)$	#Parejas
1	1486	969	0.65	2182	13.7	20
2	1542	937	0.60	2611	12.7	17
3	1337	1144	0.68	2179	11.0	10
4	2359	1012	0.72	3954	14.8	55

Puede notarse que hay una proporción directa entre la relación de repetición por oración, y el número de términos de ocurrencia unitaria normalizado con el número de parejas válidas: hay mayor productividad del método para los textos con estas características estilísticas.

También es cierto que el aumento de parejas útiles podría ser mejorado con la resolución de anáfora. En suma, con los textos de prueba utilizados hay variación de los resultados que proporciona el método frente a diversos estilos de escritura.

6. Conclusiones

Se ha aplicado una adaptación del método SEXTANT a varios textos para generar un tesauro. Asimismo, se llevó a cabo un análisis para conocer los factores que influyen en la cantidad de las parejas obtenidas. Las pruebas se realizaron con textos de diversos dominios, y de tamaño limitado por 10 000 palabras (en promedio de 6 500 palabras); lo cual es una restricción realista para aplicaciones donde se requiere información semántica del texto para, por ejemplo, hacer consultas sobre su contenido. El análisis de los resultados lleva a concluir que el método tiene sensibilidad al estilo de escritura y que es posible extraer una parte de la semántica del texto representada por parejas de términos relacionados.

Agradecimiento. Deseamos agradecer el apoyo brindado por la Coordinación de la Licenciatura en Tecnologías y Sistemas de Información de la UAM-C.

Asimismo, reconocemos la gentil colaboración de los jueces que participaron en la revisión de las parejas para poder realizar los experimentos aquí presentados, en particular al Dr. Tiburcio Moreno Olivos.

Referencias

1. Miller, G.: WordNet: A Lexical Database for English. Communications of the ACM 38, pp. 39–41 (1995)
2. Curran, J., Moens, M.: Improvements in Automatic Thesaurus Extraction. In: Proceedings of the ACL-02 Workshop on Unsupervised Lexical Acquisition, vol. 9, pp. 59–66 (2002)
3. Grefenstette, G.: Explorations on Automatic Thesaurus Discovery. Kluwer Academic Publishers (1994)
4. Rychlý, P., Kilgarriff, A.: An efficient algorithm for building a distributional thesaurus (and other Sketch Engine developments). In: Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions, Association for Computational Linguistics, pp. 41–44 (2007)
5. Jehng, J., Shihchieh, C., Cheng, C.: A Formal Concept Analysis-Based Domain-Specific Thesaurus and Its Application in Document Representation. In: Taniar, D., Gervasi, O., Murgante, B., Pardede, E., Apduhan, B. (Eds.): Computational Science and Its Applications, vol. 6018, Springer, pp. 431–442 (2010)
6. Xu, J., Croft, W.: Improving the Effectiveness of Information Retrieval with Local Context Analysis. In: ACM Transactions on Information Systems, vol. 18, ACM, pp. 79–112 (2000)
7. Lame, G.: Using NLP Techniques to Identify Legal Ontology Components: Concepts and Relations. In: Benjamins, V., Casanovas, P., Breuker, J., Gangemi, A. (Eds.): Law and the Semantic Web: Legal Ontologies, Methodologies, Legal Information Retrieval, and Applications, Springer, pp. 169–184 (2005)
8. Schmid, H.: Probabilistic Part-of-Speech Tagging Using Decision Trees. In: Proceedings of the International Conference on New Methods in Language Processing, pp. 44–49 (1994)
9. Cohen, S.: Redacción sin dolor. Planeta (2010)

Aproximaciones para la expansión semántica de consultas de un Sistema de Recuperación de Información Booleano

Ana Laura Lezama, Mireya Tovar, Darnes Vilariño, David Pinto

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

yumita1102@gmail.com, {mtovar, darnes, dpinto}@cs.buap.mx

Resumen. En el presente trabajo se proponen dos aproximaciones para la expansión de consultas de un Sistema de Recuperación de Información Booleano (SRIB), con la finalidad de mejorar los niveles de precisión de un SRIB sin expansión. Las consultas están formadas por las palabras que integran a los conceptos y las relaciones semánticas de cuatro ontologías de dominio. El propósito de estas dos aproximaciones consiste en recuperar información relevante del corpus de dominio de cada concepto y relación de la ontología de dominio. Analizando los resultados que se obtuvieron en los experimentos, se observa que la precisión del SRIB con la segunda aproximación mejora los resultados de la primera aproximación del mismo SRIB y también al SRIB sin expansión.

Palabras clave: Sistema de recuperación de información, expansión semántica de consultas, ontología.

Approaches to the Semantic Expansion of Query in a Boolean Information Retrieval System

Abstract. In this research work we propose two approaches for query expansion aiming to improve the performance of a Boolean Information Retrieval System (BIRS). Queries are made up of words and integrate the concepts and semantic relationships retrieved from four ontologies of restricted domain. The final purpose is to retrieve relevant information from the domain corpus for each concept and relationship of the ontology of restricted domain. By analyzing the results obtained in the experiments, it can be observed that the precision of the second approach proposed improves the results of the first approximation both, with and without the expansion process.

Keywords: Information retrieval system, semantic expansion of query, ontology.

1. Introducción

La Recuperación de Información *(RI)* es un campo relacionado y se centra con la estructura, almacenamiento, organización y búsqueda de elementos de información [3,12]. Tal proceso debería dar al usuario la información relevante a su necesidad de información, sin embargo, existen problemas demasiado significativos en cuanto a las consultas ingresadas por el usuario, que dificultan a un SRI recuperar toda la información relevante en cuanto a la consulta ingresada.

La función de un SRI no es la de devolver la información solicitada por el usuario, sino sólo indicar qué documentos son potencialmente relevantes para la consulta ingresada [3]. Esta investigación parte de un sistema de recuperación de información que permite recuperar documentos de un corpus de dominio, asociados a cada concepto y relaciones de una ontología de dominio. Tales conceptos y relaciones son utilizados como consultas que se emplean en la entrada a dicho sistema.

En Tovar [14] emplean un Sistema de Recuperación de Información Booleano y la información recuperada es utilizada para la evaluación automática de ontologías de dominio. Con la finalidad de mejorar la precisión del sistema mencionado, proponen la extensión del mismo. En la primera aproximación realizada para el SRIB mencionado, se extrajeron los sinónimos asociados a las consultas o conceptos completos que entran al sistema [16]. Los sinónimos son extraídos desde WordNet [7]. En la segunda aproximación las consultas están formadas por los sinónimos de cada palabra que integra a los conceptos o consultas que también son extraídos desde WordNet [7].

Esta investigación está estructurada de la siguiente manera: en la subsección 1.1 se describe la información general sobre sistemas de recuperación de información, en la sección 2 se presentan algunas propuestas por diversos autores para la expansión de consultas, en la sección 3 se puede visualizar un algoritmo general que contiene a la primera aproximación 3.1, y la segunda aproximación 3.2, en la sección 4 se presentan los experimentos y el conjunto de datos y finalmente en la sección 5 se describen las conclusiones.

1.1. Sistemas de Recuperación de Información

Los sistemas de recuperación de información, a menudo son comparados con las bases de datos relacionales. Tradicionalmente, los sistemas de recuperación de información, tienen información recuperada de textos no estructurados, lo que quiere decir que es texto en lenguaje natural. La diferencia fundamental entre bases de datos y sistemas de recuperación, es que las bases de datos son diseñadas para consultas de datos relacionales y que tienen conjuntos de archivos predefinidos, y los sistemas de recuperación de información, poseen un modelo de recuperación, un índice invertido o *postings lists*, es decir, un diccionario de términos, que nos indica el número de línea donde se encuentra el término, entre otros [6].

1.2. Sistemas de Recuperación de Información con Expansión de Consultas

La expansión de consultas o *(Query Expansion)* es la técnica comúnmente usada en Recuperación de Información, para mejorar el desempeño de los resultados por reformulación de la consulta original, ya sea añadiendo nuevos términos o reponderación de los términos originales [13].

Los términos de la expansión de consultas pueden ser automáticamente extraídos de los documentos, o tomándolos de recursos de conocimiento, como tesauros, ontologías léxicas como WordNet [7], algoritmos genéticos, etc. La ventaja de dichas técnicas es la expansión de términos que son extraídos de la colección [13], como los descritos en la sección 2.

1.3. Sistemas de Recuperación de Información sin Expansión de Consultas

Los SRI sin expansión de consultas consisten en que el usuario plasma su necesidad de información en una consulta aceptada por un SRI, por su parte el SRI transformará dicha consulta en una representación interna que permita su comparación con los documentos indexados.

La consulta supone un intento por parte del usuario de especificar las condiciones que permitan acotar dentro de la colección aquel subconjunto de documentos que contienen la información que desea. Por lo tanto, el SRI parte de la consulta formulada por el usuario, no de la necesidad de información original, por lo que una formulación incorrecta o insuficiente no podrá guiar adecuadamente al SRI durante el proceso de búsqueda. A este respecto los mayores problemas a los que ha de hacer frente el SRI son, por una parte, la escasa habilidad del usuario a la hora de formular su necesidad en forma de consulta y, por otra parte, que a la hora de describir un mismo concepto los términos empleados por el usuario y los autores de los documentos suelen diferir, impidiendo el establecimiento de correspondencias [17].

Uno de los modelos existentes, para la recuperación de información, es el modelo booleano que es uno de los métodos más utilizados para la recuperación de información [6]. Este modelo se basa en la agrupación de documentos, los cuales están compuestos por conjuntos de términos y en la concepción de las preguntas como expresiones booleanas, de ahí deriva el nombre de modelo de recuperación booleano. Es un modelo de recuperación simple, basado en la teoría de conjuntos y el álgebra booleana. Se denomina Álgebra de Boole o álgebra booleana a las reglas algebraicas, basadas en la teoría de conjuntos, para manejar ecuaciones de lógica matemática. Se denomina así en honor de George Boole, famoso matemático, que la introdujo en 1847. Dado su inherente simplicidad y su pulcro formalismo ha recibido gran atención y ha sido adoptado por muchos de los primeros sistemas bibliográficos comerciales. Su estrategia de recuperación está basada en un criterio de decisión binario (pertinente o no pertinente) sin ninguna noción de escala de medida, sin noción de un emparejamiento parcial en las condiciones de la consulta.

2. Trabajos relacionados

En el caso de la expansión de consultas, algunos autores han recurrido a diferentes técnicas de expansión, así como diferentes modelos de recuperación de información. A continuación se describen algunos trabajos relacionados con esta investigación.

En Schneider et. al. [10] plantean el uso de una ontología para mejorar los resultados de búsqueda en un SRI en un dominio en particular. Fue desarrollada dentro del marco de su investigación, que se centró en el dominio financiero. En el que se distinguen dos capas, la primera que relaciona todas la entidades presentes en el mercado bursátil y la segunda que asigna metadatos a cada una de las entidades. La expansión de la consulta fue abordada desde dos puntos de vista, lanzando la consulta completa del usuario en lenguaje natural y el análisis semántico de la consulta para expandir únicamente las entidades. El análisis semántico de la consulta se realiza utilizando Textalytics[1]. La búsqueda en lenguaje natural se basa en la utilización conjunta de la ontología y Lucene. Para la evaluación de la búsqueda en la ontología, diseñaron la prueba de Cranfield y la evaluación de cada una de las consultas es por medio de precisión y recuerdo.

En Soni et. al. [11] proponen un algoritmo genético para la expansión de consultas hechas en lenguaje natural, se utiliza el coeficiente de Czekanowski durante el proceso de expansión para que la recuperación sea más eficiente, ya que mide la similitud entre los documentos recuperados y la consulta dada. Utiliza un analizador de texto que ayuda a encontrar palabras claves en los documentos, que serán utilizadas para hacer el cromosoma que es la base del algoritmo genético. La expansión de la consulta, es realizada en base al documento que tenga el cromosoma con el mejor valor en su función de aptitud, para después hacer la expansión manualmente, usando una medida de similitud. Observaron que el uso de un algoritmo genético aumenta la relevancia de documentos recuperados, si la tasa de mutación es menor el cromosoma converge en una sola generación.

En Harb et. al. [4] usaron un rastreador que debe recorrer la WWW para obtener documentos en el dominio del cuidado de la salud, específicamente en enfermedades ictéricas. El modelo de espacio vectorial es adaptado en la propuesta de este trabajo para la representación de documentos, retira palabras vacías, etc. La consulta es expandida por sinónimos extraídos de WordNet, pero sólo con aquellos sentidos más comunes de cada término de la consulta. Con el método de recuperación de información semántico propuesto se han aprovechado las ventajas de la web semántica para recuperar documentos pertenecientes al dominio mencionado. Supera el método de recuperación de información clásica y demuestra mejoras en el rendimiento.

En Mahgoub et. al. [5] introducen una aproximación de expansión de consultas usando una ontología construída con páginas de Wikipedia, además de otros tesauros para mejorar la precisión en la búsqueda del idioma árabe. Su aproximación, depende de tres recursos árabes que son Wikipedia en árabe, como el recurso con mayor información semántica, el diccionario Al Raed, que es un dic-

[1] http://textalytics.com

cionario monolingüe para palabras modernas, y el diccionario Google_WordNet que es una colección de todas las palabras en WordNet y traducidas con el traductor de Google. La indexación y recuperación de su sistema depende de Lucene. Para expandir la consulta, primero localizan el nombre de las entidades o conceptos que aparecen en la consulta , si el nombre de una entidad o concepto es localizado, agregan el título de redirigir la página que conduce al concepto similar agregando una subcategoría del sistema. Para sus experimentos, usaron el conjunto de datos construídos desde el libro "Zad Al Ma'ad", dicho conjunto de datos contiene 25 consultas y 2,730 documentos.

En Fernández et. al. [2] muestran aspectos relacionados con la integración de la tecnología disponible del tratamiento del lenguaje natural en el desarrollo de un metabuscador que alcance un mayor grado de acierto en la recuperación de información realizada por un buscador tradicional así como en el tratamiento posterior de los documentos recuperados. Describen su proceso realizado, para la expansión de las consultas de los usuarios, con información lingüística empleando dos recursos léxicos para el castellano: ARIES que es un léxico morfológico desarrollado por la Universidad Politécnica de Madrid y la Universidad Autónoma de Madrid para el tratamiento de la morfología y EuroWordNet [18] para el tratamiento de la semántica. La generación de la consulta está compuesta por dos tareas principales, la primera consiste en transformar la consulta del usuario en lenguaje natural (LN) en una consulta formal que el buscador pueda ejecutar. La segunda funcionalidad consiste en extender los términos significativos de la consulta (formal) utilizando conocimiento lingüístico; para ello se añaden a los términos significativos de la consulta (enlazados con AND) las variantes morfológicas y semánticas mediante OR con el fin de construir una consulta en forma normal conjuntiva. Su trabajo forma parte del sistema MESIA, modelo computacional para extracción selectiva de información de textos cortos, que amplía la búsqueda habitual (consulta y presentación de resultados) con nuevas capacidades morfológicas y semánticas y analiza otros aspectos obtenidos a partir de la estructura de las páginas, del tratamiento lingüístico de algunas de las unidades de texto seleccionadas automáticamente y de la experiencia de uso.

En Cruanes et. al. [1] proponen una aproximación de mapeo de información en lenguaje natural del dominio de enfermería, utilizando métodos de similitud léxica. Los autores generan expansión por sinónimos y buscan antonímia. No usaron recursos como EuroWordNet, ya que de acuerdo a los autores, no se ajustaba a las necesidades del dominio estudiado.

En Deco et. al. [8] proponen un refinamiento semántico, que guiará al usuario a desambiguar los términos ingresados por el. Realizaron expansión semántica de consultas por sinónimos, usaron WordNet, y en la generación de la estrategia, ocuparon operadores lógicos.

En la Tabla 1 se presenta un resumen de los trabajos revisados anteriormente. En esta tabla se obsevan los recursos léxicos, los dominios, el tipo de expansión y el sistema de recuperación de información que cada autor usó en su investigación.

En esta investigación se propone la expansión de las consultas ingresadas a un SRIB, dicha expansión a diferencia de algunos trabajos del estado del arte, que

Tabla 1. Estado del arte de Sistemas de Recuperación de Información con expansión de consultas

Autores	Recursos léxicos	Dominios	Expansión de consultas	Tipo SRI
[10]	Textalytics	Financiero	Ontología	Lucene
[11]	Analizador de texto	-	Algoritmo genético	MEV
[4]	Analizador de texto	Cuidado de salud	Sinónimos	MEV
[5]	WordNet	Idioma Árabe	Ontología de dominio	Lucene
[2]	ARIES/EuroWordNet	Festivales	Sinonimia/Hiponimia	Lucene
[1]	Métodos de similitud léxica	Enfermería	Sinónimos	-
[8]	WordNet	Cuidado de la salud	Sinónimos	Booleano

realizan la expansión de consultas por medio de algoritmos genéticos, ontologías, etc. se realiza extrayendo los sinónimos de WordNet, y después usando los sinónimos extraídos, se lleva a cabo la expansión de las consultas por medio de las dos aproximaciones presentadas en este documento.

3. Propuesta

En esta sección se presenta un algoritmo general, que contiene a las dos aproximaciones de expansión propuestas, para el SRIB.

1. Extracción de conceptos y relaciones de las ontologías de dominio.
2. Extracción de los sinónimos con WordNet, esta etapa se encuentra dividida en dos procesos diferentes:
   a) Primera Aproximación
      1) Extracción de los sinónimos, de los conceptos completos con WordNet.
   b) Segunda Aproximación
      1) Se retiran palabras cerradas.
      2) Extracción de los sinónimos, de cada palabra que integran al concepto, sin palabras cerradas con WordNet.
3. Preprocesamiento del corpus de dominio, de los conceptos, de las relaciones y de los sinónimos. Esta etapa incluye las siguientes acciones:
   a) División del corpus en líneas.
   b) Eliminación de símbolos especiales, números y palabras cerradas.
   c) Aplicación de un lematizador, en particular se utiliza el algoritmo de Porter [9].
4. Formación de consultas. Existen tres tipos de consultas para las tres aproximaciones propuestas:
   a) Primera Aproximación
      1) Consultas formadas con las palabras del concepto.
      2) Consultas formadas con los sinónimos del concepto.
      3) Consultas formadas con las palabras del concepto que forman la relación semántica.
   b) Segunda Aproximación

   1) Consultas formadas con las palabras del concepto.
   2) Consultas formadas con los sinónimos de cada palabra que integra al concepto, y los conceptos originales.
   3) Consultas formadas con las palabras del concepto que forman la relación semántica.
5. Aplicación del Sistema de Recuperación de Información Booleno (SRIB) para conceptos, sin expansión.
6. Aplicación del Sistema de Recuperación de Información Booleno (SRIB) para los sinónimos de los conceptos, con expansión.
7. Mezcla o unión de los resultados obtenidos (posting) por el SRIB para resultados sin sinónimos y con sinónimos de los dos pasos anteriores.
8. Aplicación del operador AND para la consulta que incluye los dos conceptos que forman la relación semántica. El operador AND realiza la intersección de las líneas que integran los posting de ambos conceptos que forman la relación semántica.

En el caso de la evaluación de los resultados obtenidos, se utilizan las Ecuaciones (1) y (2) para medir la precisión a nivel de conceptos y relaciones:

$$P_C = \frac{Conceptos\ recuperados}{Total\ conceptos}, \tag{1}$$

$$P_R = \frac{Relaciones\ recuperadas}{Total\ relaciones}, \tag{2}$$

donde *Conceptos recuperados* es el total de conceptos obtenidos por el SRIB, y el *Total conceptos* es el total de conceptos existentes en la ontología de dominio. En el caso de *Relaciones recuperadas* se evalúa por separado las relaciones taxonómicas y las relaciones no taxonómicas (para más información ver [15]). El *Total relaciones* corresponden al total de relaciones de cada tipo existentes en la ontología de dominio evaluadas de manera independiente. A continuación se presenta brevemente cada aproximación propuesta.

3.1. Primera aproximación

La primera aproximación, realiza la extracción de sinónimos de los conceptos o consultas, completos de cada ontología de dominio, y después se hace uso del algoritmo de unión o mezcla de los documentos recuperados por el SRIB sin expansión y con expansión, y posteriormente la evaluación del mismo. En la Figura 1 se muestra el comportamiento de manera gráfica del algoritmo general, incluyendo únicamente la primera aproximación.

3.2. Segunda aproximación para la expansión de consultas

En la segunda aproximación se plantea la expansión de la consulta, al incorporar los sinónimos correspondientes a cada palabra que forman al concepto de la ontología. En la Figura 2 se observa de manera gráfica los pasos que

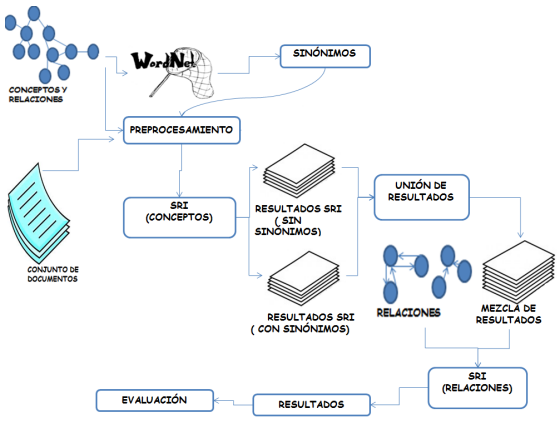

Fig. 1. Primera aproximación para la expansión de consultas en un SRIB

se siguieron en el algoritmo general incluyendo sólo la segunda aproximación. La figura incluye la búsqueda de los sinónimos de cada palabra que integran al concepto (sin cerradas en WordNet), así como la generación de las nuevas consultas procesadas por el SRIB, la unión de los resultados del SRIB con expansión y sin expansión.

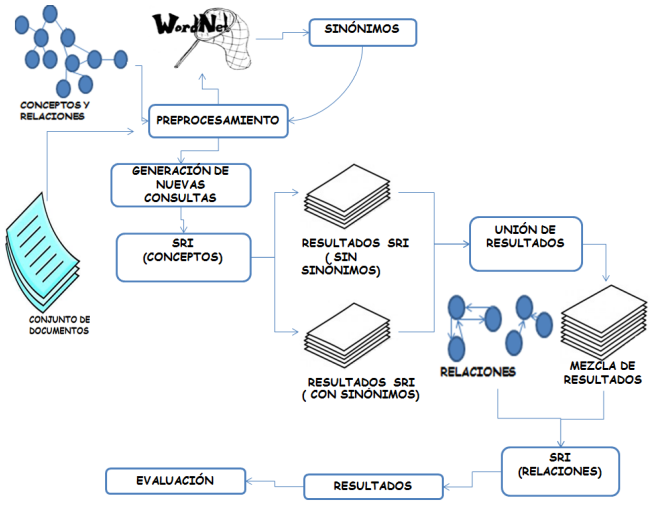

Fig. 2. Segunda aproximación para la expansión de consultas en un SRIB

4. Resultados experimentales

En esta sección, se presentan los datos utilizados (4.1) y los resultados obtenidos en los experimentos (4.2).

4.1. Conjunto de datos

En la Tabla 2 se presenta el número de conceptos (C), el total de relaciones taxonómicas (T) y el total de relaciones no taxonómicas (NT) de las ontologías evaluadas. También se incluye el número de documentos (D), el número de tokens (T), la cantidad de vocabulario (V), y el número de oraciones. Los dominios utilizados en los experimentos son Inteligencia Artificial (IA), Aprendizaje e-Learning (SCORM) [19], ontología del dominio de Petróleo (OIL), y Turismo (Turismo).

Tabla 2. Conjunto de datos

Dominio	Ontología			Corpus de referencia			
	C	T	NT	D	T	V	O
AI	276	205	61	8	11,370	1,510	475
SCORM	1,461	1,038	759	36	1,621	34,497	1,325
OIL	48	37	-	577	546,118	10,290,107	168,554
Turismo	963	1,016	-	1,801	877,519	32,931	36,505

4.2. Resultados obtenidos

A continuación se presentan los resultados experimentales obtenidos por los dos algoritmos desarrollados y su comparación, es decir, resultados del Sistema de Recuperación de Información Booleno (SRIB) sin expansión de consultas y del Sistema de Recuperación Información Booleano (SRIB) con expansión de consultas, para las dos aproximaciones presentadas.

Primera aproximación Se presentan los resultados experimentales obtenidos por los dos algoritmos desarrollados y su comparación, es decir, resultados del Sistema de Recuperación de Información Booleno (SRIB) sin expansión de consultas y del Sistema de Recuperación Información Booleano (SRIB) con expansión de consultas. Los resultados obtenidos por ambos algoritmos, para el caso de los conceptos, se muestran en la Tabla 3 para cada ontología revisada (Dominio). En la Tabla 3 también se muestra el total de conceptos extraídos de la ontología (CO), los conceptos recuperados por el SRIB sin expansión (C), los conceptos que no obtuvieron líneas asociadas (F) y la precisión (P); los conceptos recuperados por el SRIB con expansión (CE), los conceptos que no logró recuperar el SRIB con expansión (FE) y la precisión obtenida (PE).

Además, en la tabla se incluye la cantidad de oraciones obtenidas por el SRIB sin expandir (O), con expansión (OE), la diferencia del número de líneas recuperadas con expansión y sin ella (DI) y el porcentaje de incremento (%). En base a los resultados obtenidos para los conceptos, se observa que en los casos de los dominios de SCORM y Turismo principalmente, se incrementó el número de conceptos recuperados que los que se recuperan con el SRIB sin expansión. Además, la cantidad de oraciones que contienen los sinónimos del concepto incrementa la cantidad de líneas u oraciones asociadas a cada concepto de las ontologías, esto ocurre para cada dominio. El porcentaje de incremento de la información recuperada en número de líneas, por el SRIB con expansión es mayor al 27 %, lo que indica que el concepto puede ser representado en el corpus por su sinónimo correspondiente y que esta información es adicional a la presentada por el SRIB sin expansión.

Tabla 3. Resultados de la primera aproximación, del Sistema de Recuperación Booleano con expansión para el caso de los conceptos de cada ontología de dominio

Dominio	Ontología							Corpus			
	CO	C	F	P	CE	FE	PE	O	OE	DI	%
IA	276	274	2	0.992	274	2	0.992	1,994	3,057	1,063	53.30
SCORM	1,461	1,434	27	0.981	1,435	26	0.982	23,406	31,093	7,687	32.84
OIL	48	48	0	1.00	48	0	1.00	232,603	295,986	63,383	27.24
Turismo	963	682	281	0.708	711	252	0.738	86,353	224,764	138,411	160.28

En la Tabla 4 se presentan los resultados obtenidos por ambos Sistemas de Recuperación de Información con expansión y sin ella, para relaciones de tipo taxonómicas de cada ontología de dominio. La columna RT corresponde al total de relaciones taxonómicas incluidas en la ontología de dominio correspondiente. La columna RR es el total de relaciones recuperadas con información del SRI sin expansión y con expansión RRE. La columna correspondiente a F es la diferencia de las relaciones recuperadas por el SRI booleano sin expansión y con expansión (FE). La precisión del sistema sin expansión (P) y con expansión (PE). También se incluye la cantidad de oraciones recuperadas en total por el SRIB sin expansión (O) y con expansión (OE) para este tipo de relaciones, la diferencia obtenida (DI) y el porcentaje de la diferencia (%).

En base a los resultados obtenidos se observa que el número de relaciones de tipo taxonómicas de las tres primeras ontologás se mantienen por los dos algoritmos diseñados, pero en el caso de la ontología de Turismo el número de relaciones se incrementa de 291 a 386 esto indica que existen relaciones en el corpus que sólo se pueden encontrar por su correspondiente sinónimo y al SRIB sin expansión no le es posible encontrarlo exactamente. También, la cantidad de oraciones asociadas a los SRIB con expansión se incrementa para las cuatro ontologías y más aún para la ontología de Turismo, reforzando nuevamente la existencia de los sinónimos de las relaciones encontradas en el corpus.

Tabla 4. Resultados de la primera aproximación, del Sistema de Recuperación Booleano con expansión para el caso de las relaciones taxonómicas de cada ontología de dominio

Dominio	Ontología							SRI			
	RT	RR	F	P	RRE	FE	PE	O	OE	DI	%
IA	205	205	0	1.00	205	0	1.00	782	876	94	12.02
SCORM	1,038	1,002	36	0.965	1,002	36	0.965	10,640	10,926	286	2.68
OIL	37	32	5	0.864	32	5	0.864	12,696	12,704	8	0.063
Turismo	1,016	291	725	0.286	386	630	0.379	5,606	20,198	14,592	260.29

En el caso de las relaciones tipo no taxonómicas, que sólo las ontologías IA y SCORM tienen, se observa que la cantidad de relaciones recuperadas es la misma para ambos sistemas. La columna RNT corresponde, al número de relaciones no taxonómicas incluídas en cada ontología de dominio correspondiente. Las columnas R y RE, corresponde al número de relaciones obtenidas por el SRI sin expansión, y con expansión, respectivamente. Las columnas FE y F, corresponden a la diferencia de las relaciones recuperadas por el SRI con expansión y sin expansión, respectivamente. Las columnas P y PE, corresponde a la precisión del sistema sin expansión y con expansión respectivamente. Las columnas O y OE, corresponde a la cantidad de oraciones recuperadas por el SRI sin expansión y con expansión respectivamente y la columna DI, corresponde a la diferencia de oraciones recuperadas del sistema con expansión y el sistema sin expansión. En este caso, sólo se incrementaron algunas oraciones en las cuales existen el sinónimo correspondiente a cada concepto que forma la relación (ver Tabla 5).

Tabla 5. Resultados de la primera aproximación, para relaciones no taxonómicas

Dominio	Ontología							SRI			
	RNT	R	F	P	RE	FE	PE	O	OE	DI	%
IA	61	61	0	1.000	61	0	1.000	108	136	28	25.92 %
SCORM	759	738	21	0.972	738	21	0.972	8,728	9,655	927	10.62 %

Segunda aproximación A continuación se presentan los resultados experimentales obtenidos por los dos algoritmos desarrollados y su comparación, es decir, resultados del Sistema de Recuperación de Información Booleno (SRIB) sin expansión de consultas y del Sistema de Recuperación Información Booleano (SRIB) con expansión de consultas, con la diferencia de la primera aproximación de que los conceptos o consultas ingresados en el SRIB, son ahora nuevas consultas creadas con los sinónimos extraídos de WordNet. Los resultados obtenidos por ambos algoritmos, para el caso de los conceptos, se muestran en la Tabla 6

para cada ontología revisada (Dominio), siguiendo la nomenclatura utilizada en la Tabla 3.

En base a los resultados obtenidos para los conceptos, se observa que en los casos de los dominios de SCORM y Turismo principalmente, se incrementó el número de conceptos recuperados que los que se recuperan con el SRIB sin expansión. Además, la cantidad de oraciones que contienen los sinónimos del concepto incrementa la cantidad de líneas u oraciones asociadas a cada concepto de las ontologías, esto ocurre para cada dominio. El porcentaje de incremento de la información recuperada por el SRIB con expansión es mayor al 33 %, lo que indica que el concepto puede ser representado en el corpus por su sinónimo correspondiente y que esta información es adicional a la presentada por el SRIB sin expansión.

Tabla 6. Resultados de la segunda aproximación, del Sistema de Recuperación Booleano con expansión para el caso de los conceptos de cada ontología de dominio

Dominio	Ontología							Corpus			
	CO	C	F	P	CE	FE	PE	O	OE	DI	%
IA	276	274	2	0.992	274	2	0.992	1,994	3,325	1,331	66.75
SCORM	1,461	1,434	27	0.981	1,436	25	0.982	23,406	35,987	12,581	53.75
OIL	48	48	0	1.00	48	0	1.00	232,603	310,067	77,464	33.30
Turismo	963	683	281	0.708	784	179	0.814	86,353	227,451	141,098	163.39

En la Tabla 7 se presentan los resultados obtenidos por ambos Sistemas de Recuperación de Información con expansión y sin ella, para relaciones de tipo taxonómicas de cada ontología de dominio. En base a los resultados obtenidos se observa que el número de relaciones de tipo taxonómicas de las dos primeras ontologías se mantienen por los dos algoritmos diseñados, pero en el caso de la ontología de OIL el número de conceptos aumenta en uno, mientras que para la ontología de Turismo el número de conceptos se incrementa de 291 a 433 esto indica que existen conceptos en el corpus que sólo se pueden encontrar por su correspondiente sinónimo y al SRIB sin expansión no le es posible encontrarlo exactamente. También, la cantidad de oraciones asociadas a los SRIB con expansión se incrementa para las cuatro ontologías y más aún para la ontología de Turismo, reforzando nuevamente la existencia de los sinónimos de los conceptos encontrados en el corpus.

En el caso de las relaciones tipo no taxonómicas, que sólo las ontologías IA y SCORM tienen, se observa que la cantidad de relaciones recuperadas es la misma para ambos sistemas. Sólo se incrementaron oraciones en las cuales existen el sinónimo correspondiente a cada concepto que forma la relación (ver Tabla 8).

Tabla 7. Resultados de la segunda aproximación, del Sistema de Recuperación Booleano con expansión para el caso de las relaciones taxonómicas de cada ontología de dominio

Dominio	Ontología							SRI			
	RT	RR	F	P	RRE	FE	PE	O	OE	DI	%
IA	205	205	0	1.00	205	0	1.00	782	1089	307	39.25
SCORM	1,038	1,002	36	0.965	1,002	36	0.965	10,640	14,498	3,858	36.25
OIL	37	32	5	0.864	33	4	0.891	12,696	18,736	6,040	47.57
Turismo	1,016	291	725	0.286	433	583	0.426	5,606	22,823	17,217	307.11

Tabla 8. Resultados de la segunda aproximación, para relaciones no taxonómicas

Dominio	Ontología							SRI			
	RNT	R	F	P	RE	FE	PE	O	OE	DI	%
IA	61	61	0	1.000	61	0	1.000	108	149	41	37.96 %
SCORM	759	738	21	0.972	738	21	0.972	8,728	10,262	1,534	17.57 %

5. Conclusiones y trabajo futuro

En este artículo se presentan dos aproximaciones para la expansión de consultas en un Sistema de Recuperación de Información Booleano. La primera aproximación consiste en expandir utilizando los sinónimos del concepto exacto. La segunda aproximación realiza la expansión de consultas con el uso de sinónimos de cada palabra que integra a los conceptos. Las consultas están formadas por los conceptos extraídos de las ontologías de dominio. De acuerdo a los resultados experimentales se visualiza que la expansión de la segunda aproximación permite recuperar más información del corpus de dominio, en comparación con el SRIB sin expansión y con la primera aproximación realizada. En algunos casos el SRIB con expansión utilizando la segunda aproximación permite recuperar más conceptos, relaciones e información asociada a estos conceptos desde el corpus, al incorporar los sinónimos de las palabras que conforman a los conceptos desde WordNet. En algunas ontologías la cantidad de oraciones recuperadas supera significativamente al SRIB sin expansión, y a la primera aproximación. Como trabajo a futuro se propone el uso de patrones léxico-sintácticos para la extracción de relaciones tipo sinonimía desde el corpus de dominio, que permitan identificar los sinónimos de los conceptos de las ontologías de dominio.

Agradecimientos. Este trabajo de investigación esta parcialmente financiado por el número de proyecto VIEP 302 (2016), por el proyecto PRODEP (EXB-792) con número de convenio DSA/103.5/15/10854.

Referencias

1. Cruanes, J., Ferri, M.T.R., Pastor, E.L.: Análisis del uso de métodos de similitud léxica con conocimiento semántico superficial para mapear la información de enfermería en español. Procesamiento del lenguaje natural 49, 75–82 (2012)
2. Fernández, P.M., Serrano, A.G.: Utilizando recursos lingüísticos para mejora de la recuperación de información en la web. Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial 6(16), 55–64 (2002)
3. Ferro, J.V., Nistal, J.L.F.: Aplicaciones del procesamiento del lenguaje natural en la recuperación de información en español. Procesamiento del Lenguaje Natural 36, 57–58 (2006)
4. Harb, H.M., Fouad, K.M., Nagdy, N.M.: Semantic retrieval approach for web documents. International Journal of Advanced Computer Science and Applications (IJACSA) 2(9), 67–76 (2011)
5. Mahgoub, A.Y., Rashwan, M.A., Raafat, H., Zahran, M.A., Fayek, M.B.: Semantic query expansion for arabic information retrieval. In: EMNLP: The Arabic Natural Language Processing Workshop, Conference on Empirical Methods in Natural Language Processing, Doha, Qatar. pp. 87–92 (2014)
6. Manning, C.D., Raghavan, P., Schütze, H., et al.: Introduction to information retrieval, vol. 1. Cambridge university press Cambridge (2008)
7. Miller, G.A.: Wordnet: A lexical database for english. Communications of the ACM 38(11), 39–41 (1995)
8. Motz, R., Deco, C., Bender, C., Saer, J., Chiari, M.: Refinamiento semántico para recuperación de información desde la web. In: Proceedings Workshops on Artificial Intelligence, Iberamia. pp. 172–179 (2004)
9. Porter, M.F.: Readings in information retrieval. chap. An Algorithm for Suffix Stripping, pp. 313–316. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1997)
10. Schneider, J.M., Declerck, T., Fernández, J.L.M., Martínez, P.: Prueba de concepto de expansión de consultas basada en ontologías de dominio financiero. Procesamiento del lenguaje natural 51, 109–116 (2013)
11. Soni, N., Singh, J.: Relevancy enhancement of query with czekanowski coefficient by expanding it using genetic algorithm. International Journal of Computer Science and Information Technologies 5, 6106–6110 (2014)
12. Tolosa, G.H., Bordignon, F.R.: Introducción a la recuperación de información pp. 1–149 (2008)
13. Vechtomova, O., Wang, Y.: A study of the effect of term proximity on query expansion. Journal of Information Science 32(4), 324–333 (2006)
14. Vidal, M.T.: Evaluación automática de ontologías de dominio restringido. Ph.D. thesis, Cenidet (Febrero 2015)
15. Vidal, M.T., Avendaño, D.P., Rendón, A.M., Serna, J.G.G., Ayala, D.V.: Evaluation of ontological relations in corpora of restricted domain. Computación y Sistemas 19(1), 135–149 (2015)
16. Vidal, M.T., Sánchez, A.L.L., Ayala, D.V.n., Beltrán Martínez, B., Cardona, M.C.: Primera aproximación de un sistema de recuperación de información booleano con expansión semántica de consultas. Research in Computing Science 99, 55–63 (2015)
17. Vilares Ferro, J.: Aplicación del procesamiento del lenguaje natural en la recuperación de información en español. Ph.D. thesis, Universidad da Coruña, Departamento de Computación (Mayo 2005)

18. Vossen, P.: Introduction to eurowordnet. Computers and the Humanities 32(2), 73–89 (1998)
19. Zouaq, A., Gasevic, D., Hatala, M.: Linguistic patterns for information extraction in ontocmaps. In: Blomqvist, E., Gangemi, A., Hammar, K., del Carmen Suárez-Figueroa, M. (eds.) WOP. CEUR Workshop Proceedings, vol. 929. CEUR-WS.org (2012)

Agrupamiento de textos cortos en dominios cruzados

Alba Núñez-Reyes[1], Erick Monroy-Cuevas[1], Esaú Villatoro-Tello[2], Gabriela Ramírez-de-la-Rosa[2], Christian Sánchez-Sánchez[2]

[1] Universidad Autónoma Metropolitana (UAM) Unidad Cuajimalpa, Maestría en Diseño, Información y Comunicación (MADIC), División de Ciencias de la Comunicación y Diseño, México

[2] Universidad Autónoma Metropolitana (UAM) Unidad Cuajimalpa, Departamento de Tecnologías de la Información, México

{ar.nunezreyes, ermoncu}@gmail.com, {gramirez,evillatoro,csanchez}@correo.cua.uam.mx

Resumen. Recientemente, las redes sociales se han vuelto un medio ideal para compartir información en tiempo real. Diversos tipos de usuarios las emplean para comentar sus gustos, actividades u opiniones. Esta información vertida en estos medios se ha vuelto de particular interés para los analistas de reputación en línea, pues a través de éstas logran identificar tendencias relevantes. Sin embargo, analizar miles de datos se vuelve una tarea tediosa para el humano. Existen técnicas de clasificación de documentos las cuales representan soluciones alternativas al problema anterior, pero dada la dinámica de las redes sociales, el tener un modelo de clasificación de documentos por tendencias se vuelve una tarea impensable, pues al surgir nuevas temáticas o al cambiar de dominio de análisis, los modelos construidos no se desempeñarán eficientemente. En este trabajo presentamos un método no supervisado para la identificación de temáticas en textos cortos. Nuestros resultados experimentales muestran que el método propuesto permite tener una representación de textos robusta, que se comporta satisfactoriamente en dominios cruzados.

Palabras clave: Agrupamiento de documentos, dominios cruzados, representación de documentos, selección de atributos, coeficiente de Silhouette.

Cross-domain Clustering for Short Texts

Abstract. Nowadays, social networks have become an ideal tool for sharing information in real time. Different type of users use social media to comment about their activities, opinions, personal views, etc. The information poured in this media has become of particular interest to online reputation analysts, for instance, to identify relevant tendencies.

However, the analysis of great amount of data is a very tedious task for a human. There are classification techniques that present alternative solutions for this problem, but given the dynamism of these social networks, having a model for each trend is not feasible since every day are emerging new trends, and even worse, new trends in new domains. In this paper, we present an unsupervised method for short texts categorization. Our experimental results show that our proposed method allows a robust text representation that performs well in cross-domains problems.

Keywords: Document clustering, cross-domain, document representation, feature selection, Silhouette coefficient.

1. Introducción

Actualmente algunos esfuerzos de investigación en materia de Procesamiento del Lenguaje Natural se han enfocado en la aplicación de técnicas de agrupamiento de datos para identificar las temáticas de documentos cortos, en particular de tuits [13]. Twitter se ha convertido en una herramienta de micro-blogging que permite dar seguimiento, en tiempo real, a gran diversidad de eventos que suceden alrededor del mundo. La información vertida en esta red social se ha vuelto de interés para distintos sectores, *e.g.*, económico, académico, político, comercial y empresarial; pues la información contenida en los 140 caracteres permiten a éstos posicionar opiniones o definir tendencias.

El Analista de Reputación en Línea (ARL) es la persona encargada de analizar opiniones e identificar tendencias respecto a una figura pública y/o entidad de interés. Una vez que el ARL logra identificar información que le es relevante, es capaz de sugerir estrategias de mercado, de tal manera que pueda invertir o potenciar determinadas posturas. El principal problema que enfrenta la figura del ARL es la imperiosa necesidad de analizar miles de textos (tuits) para poder realizar de manera eficiente su labor. Dado esto, en el año 2012 se propone por primera vez un marco de evaluación, llamado RepLab [3,1,2], cuyo principal objetivo es impulsar el desarrollo de sistemas automáticos que apoyen en las actividades de un ARL.

Entre los retos propuestos en el RepLab, se propuso la tarea de identificación de tópicos (*topic detection*), la cual consiste en desarrollar métodos que sean capaces de agrupar tuits relacionados por un tópico en común, con el objetivo de permitir al ARL obtener conjuntos de tuits que refieren a la misma temática. En general, el agrupamiento de datos se define como la tarea de construir grupos de objetos, de tal manera que los elementos de un mismo grupo sean muy similares entre si, pero diferentes a los elementos de otro grupo [6], propiedades conocidas como homogeneidad y heterogeneidad respectivamente.

En el área de Procesamiento de Lenguaje Natural se han propuesto diversas estrategias de agrupamiento que funcionan muy bien cuando se trata de documentos formales (*e.g.*, artículos, noticias, libros, etc.) y que además

emplean técnicas de aprendizaje supervisadas, es decir, tienen un conjunto de entrenamiento etiquetado (una clase por tópico) que permite la construcción de modelos de clasificación confiables [14]. Sin embargo, en un contexto en el que los tópicos son muy diversos, e incluso desconocidos, el contar con datos etiquetados se vuelve una práctica muy costosa y frecuentemente inimaginable.

Con la finalidad de resolver los problemas mencionados, en este trabajo proponemos un método de clasificación no supervisado (*i.e.*, agrupamiento) en el cual se empleó una forma de representación compacta de los textos. Nuestra hipótesis es que la alta dimensionalidad de técnicas tradicionales de representación de documentos afecta el comportamiento de algoritmos de agrupamiento cuando se trata de documentos cortos y además de dominios distintos. En este sentido, una representación compacta, la cual elimina términos muy especializados (dependientes del dominio) y términos muy comunes, permitirá generar un agrupamiento temático más confiable para dominios cruzados.

Para realizar nuestros experimentos trabajamos con una muestra de los datos proporcionados por los organizadores del RepLab del 2013[3]. Los resultados obtenidos muestran que la forma de representación propuesta para realizar el agrupamiento de dominio cruzado, permite obtener un comportamiento similar al que se logra cuando se trabaja en un esquema de "in-domain", es decir, cuando se evalúa el método en el mismo dominio en el que fue construida la representación.

El resto de este documento se encuentra organizado de la siguiente manera. En la sección 2 se describen algunos de los trabajos relacionados al problema de agrupamiento de textos cortos. En la sección 3 se describe en detalle nuestro método propuesto para el agrupamiento de documentos en un escenario de dominio cruzado. En la sección 4 se describe la metodología experimental y los resultados obtenidos. Finalmente, en la sección 5 se plantean las conclusiones obtenidas y se describen algunas lineas de trabajo futuro.

2. Trabajo relacionado

El reto al momento de trabajar con documentos cortos es, principalmente, que la estructura de los textos cortos no sigue las convenciones léxicas y sintácticas de la mayoría de documentos formales, razón por lo cual se presentan dificultades con métodos tradicionales de agrupamiento. En consecuencia, la forma de representación de textos cortos se ha convertido en un área de interés para la comunidad científica, sobre todo cuando se quieren proponer representaciones que sean robustas a cambios de dominio, donde se sabe que la distribución de los atributos será distinta entre dominios. Tradicionalmente, la forma de representación de textos empleada por técnicas de agrupamiento es la bolsa de palabras (BOW). Esta consiste en representar a un documento como el conjunto total de las palabras que aparecen en él, no obstante, esta técnica tiende a ignorar las relaciones semánticas entre las palabras, por lo cual se pierde gran parte del significado de un documento.

[3] `http://nlp.uned.es/replab2013/`

En el trabajo descrito en [8] los autores proponen eliminar las limitaciones de la representación BOW por medio de enriquecer el texto haciendo uso de ontologías. La idea intuitiva de este enfoque consiste en identificar relaciones entre conceptos de los artículos de Wikipedia y los términos presentes en la bolsa de palabras de cada documento; una vez identificados, los últimos son considerados para enriquecer el texto del documento en revisión. De manera similar, en [4] se propone un método de agrupamiento de notas extraídas de servicios RSS[4] y snippets[5] de Google, el cual también busca enriquecer la BOW empleando Wikipedia, para tener un agrupamiento más efectivo.

En el trabajo descrito en [10], los autores proponen un método de agrupación de textos cortos por medio de la identificación de términos "núcleo". En esencia, el método propuesto es un proceso iterativo que identifica el término núcleo de un conjunto de textos cortos y realiza un primer agrupamiento. Posteriormente, el proceso de identificación de términos núcleo se repite y genera una nueva propuesta de agrupamiento. Este proceso se repite hasta que un criterio de paro es alcanzado, el cual se basa en una medida de calidad de los grupos formados.

Por otro lado, un trabajo que intenta incorporar información secuencial por medio del uso de n-gramas de palabras es el descrito en [7]. Esta propuesta es evaluada en documentos cortos escritos en Mandarín. Los autores argumentan que esta forma de representación permite capturar, además de información léxica, información sobre la estructura de los textos y algunos aspectos semánticos.

Finalmente, en el trabajo descrito en [16] el problema de agrupamiento en dominio cruzado es aproximado por medio de identificar atributos "crudos" que incorporan, en la representación de los textos tanto aquellos atributos que son compartidos por ambos dominios, como atributos que son completamente disjuntos. Para lograr esta incorporación, los autores utilizan un método probabilístico (EM). A pesar de que este trabajo enfrenta el mismo problema que nosotros queremos resolver, su método requiere de una etapa de entrenamiento, lo cual lo vuelve dependiente de los datos etiquetados disponibles así como dependiente del dominio.

En los trabajos descritos previamente, se pueden identificar las siguientes desventajas: *i)* el problema del agrupamiento en dominio cruzado no es contemplado como problema primario, *ii)* los que enfrentan el problema de dominio cruzado dependen de la existencia de datos etiquetados, y *iii)* la forma de representación propuestas son de muy alta dimensionalidad. Así entonces, en este artículo proponemos usar una forma de representación compacta de los documentos, la cual no considera a elementos dependientes del dominio y al mismo tiempo elimina términos muy comunes. Note que estos dos tipos de atributos pueden ser los causantes de ruido al momento de hacer el agrupamiento.

[4] Siglas de Really Simple Syndication, un formato XML para compartir contenido en la web.

[5] Término extraído del idioma Inglés que refiere a pequeños fragmentos de texto.

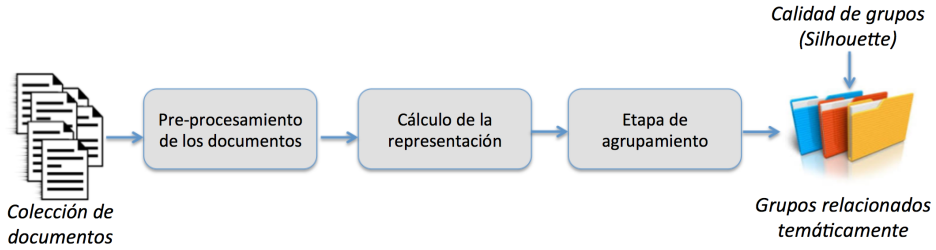

Fig. 1. Arquitectura general de un método de agrupamiento

3. Método propuesto

Como se ha mencionado antes, en el caso del agrupamiento como tarea no supervisada, el problema es encontrar y formar grupos significativos a partir de una colección de elementos no etiquetados. En cierta forma, las etiquetas son asociadas con el número de grupos, la característica de estas etiquetas es que son deducidas únicamente a partir de los elementos de entrada (Figura 1).

Como se puede observar en la Figura 1, los pasos involucrados en la tarea de agrupamiento de documentos son: *1)* pre-procesamiento de la información, esto es, la eliminación de los elementos que podrían causar ruido al momento de realizar el agrupamiento, por ejemplo etiquetas HTML, palabras funcionales, números, signos de puntuación, etc.; *2)* construcción de la representación de los documentos y al mismo tiempo la definición de una medida de proximidad apropiada al conjunto de datos; *3)* aplicación de un algoritmo de agrupamiento; y *4)* la evaluación sobre la calidad de los grupos formados.

Es importante mencionar que en un esquema de agrupamiento de dominio cruzado, el principal objetivo de este trabajo, el reto está en identificar una forma de representación robusta que permita este cambio temático de la colección de documentos entre dominios. Por lo tanto, se busca que se generen grupos de calidad en dominios distintos. Así, nuestro principal interés está en la evaluación de distintas formas de representación para el problema de dominios cruzados. A continuación describimos en detalle cómo se abordaron cada uno de los pasos descritos en la Figura 1.

3.1. Pre-procesamiento de los tuits

Previo al proceso de representación de los documentos se realizó un preprocesamiento a los tuits el cual consistió en los siguientes pasos:

1. Los tuits se convierten a minúsculas con la finalidad de normalizar el vocabulario.
2. Cualquier secuencia de espacios en blanco se convierte en un solo espacio.
3. Se eliminaron las menciones a usuario (*@usuario*) así como cualquier URL que existiera en los tuits.

4. Se eliminan los signos de puntuación. Esto también elimina cualquier emoticono que pudiera aparecer, puesto que no fueron tomados en cuenta para el funcionamiento de este modelo.
5. Cada palabra de un tuit es llevada a su raíz léxica. Este proceso se lleva a cabo mediante el lematizador Porter [11].
6. Se eliminan las palabras vacías y/o funcionales.

3.2. Método base para la representación de los documentos

Una vez eliminados los elementos considerados ruidosos en la etapa de preprocesamiento, el paso obligado es el *indexado* de los documentos de entrenamiento (T), actividad que denota hacer el mapeo de un documento d_j en una forma compacta de su contenido. La representación más comúnmente utilizada para representar cada documento es un vector con términos ponderados como entradas, concepto tomado del modelo de espacio vectorial usado en recuperación de información. Esto es, cada texto d_j es representado como el vector $\overrightarrow{d}_j = \langle w_{kj}, \ldots, w_{|\tau|j} \rangle$, donde τ es el *diccionario*, *i.e.*, el conjunto de términos que ocurren al menos una vez en algún documento de T, mientras que w_{kj} representa la importancia del término t_k dentro del contenido del documento d_j. En ocasiones τ es el resultado de filtrar las palabras del vocabulario, *i.e.,* resultado de un pre-procesamiento (Sección 3.1). Una vez que hemos hecho los filtrados necesarios, el diccionario τ puede definirse de acuerdo a diferentes criterios, sin embargo el que se empleó como método base en esta propuesta corresponde a la Bolsa de Palabras (BOW).

Los pesos w_{kj} pueden ser definidos de variadas formas, sin embargo la que nosotros empleamos es la de ponderado booleano; este pesado consiste en asignar el peso de 1 si la palabra ocurre en el documento y 0 en otro caso. La razón principal para seleccionar un esquema de pesado booleano es debido a la misma naturaleza de los documentos, *i.e.,* tuits. Ya que los tuits son documentos muy cortos (140 caracteres) consideramos que el uso de frecuencias no incorporaría información relevante al algoritmo de agrupamiento.

3.3. Método propuesto para la representación de los documentos

La representación tradicional BOW trae un costo agregado, que es el producir un espacio de términos (atributos) τ de alta dimensionalidad (*i.e.*, $|\tau| \to \infty$). Esta alta dimensionalidad puede ocasionar problemas de *sobre-ajuste* en el proceso de agrupamiento, *i.e.,* el fenómeno por el cual un método de agrupamiento se adapta a las características contingentes de T, en lugar de únicamente a las características constitutivas de las categorías; esto provoca problemas de efectividad pues el algoritmo de agrupamiento tiende a comportarse mejor sobre los datos con los que ha sido evaluado, sin conservar esta tendencia en conjuntos de datos distintos (*i.e.,* dominios diferentes).

Uno de los métodos que ha mostrado ser efectivo como técnica de reducción de dimensionalidad es aquel que conserva sólo los términos que se encuentran

alrededor del punto de transición (pt_T) [5]. El pt_T es un valor de frecuencia que divide a los términos del vocabulario τ en dos conjuntos de términos, los de baja y alta frecuencia [17,5]. En los estudios realizados en [17,5] se demuestra que los términos de frecuencia media están fuertemente relacionados con el contenido de los documentos, lo cual permite resolver efectivamente tareas como el agrupamiento temático de textos. Nuestra hipótesis es que por medio de utilizar esta representación compacta, la cual considera términos de frecuencias medias a altas, es posible generar un modelo de agrupamiento robusto al cambio de dominio. La forma tradicional de calcular el punto de transición es:

$$tp_T = \frac{\sqrt{8 * I_1 + 1} - 1}{2}, \tag{1}$$

donde I_1 representa el número de palabras con frecuencia 1 en el vocabulario τ.

3.4. Medida de proximidad

Para todos los experimentos realizados se utilizó a la medida del *coseno* como métrica de proximidad. El objetivo de esta métrica es contar con un valor numérico al cual llamaremos coeficiente de similitud SC, el cual nos dirá cuán parecidos son los documentos d_i y d_j dados. La idea básica de la medida del coseno es determinar el ángulo entre el vector de d_i y de d_j, para hacerlo, calculamos:

$$SC(d_i, d_j) = \frac{\sum_{k=1}^{t} w_{ik} w_{jk}}{\sqrt{\sum_{k=1}^{t} (w_{jk})^2 \sum_{k=1}^{t} (w_{ik})^2}}, \tag{2}$$

donde k va de 1 al número total de términos del vocabulario τ, w_{ik} indica la importancia del término k en el documento d_i mientras que w_{jk} indica la importancia del término k en el documento d_j.

3.5. Algoritmos de agrupamiento

El agrupamiento de documentos consiste en que dado un conjunto de N documento, $\mathcal{D}$, se busca dividirlos o particionarlos en un número predeterminado de k subconjuntos tal que $\mathcal{D} = \bigcup_{i=1}^{k} \mathcal{D}_i$, de tal forma que los documentos asignados a cada subconjunto sean más similares entre sí que a los documentos asignados a otros subconjuntos, *i.e.*, minimizar la distancia intra-clusters o maximizar la semejanza intra-clusters [9].

En este trabajo utilizamos dos métodos de agrupamiento que han mostrado buen desempeño en agrupamiento de documentos de textos: algoritmo de agrupamiento estrella y algoritmo de agrupamiento *k-means*. Ambos algoritmos se describen brevemente a continuación.

Algoritmo estrella. Este es un algoritmo dinámico que induce de manera natural el número de grupos a formar y la estructura de los temas dentro del espacio de textos. Es un algoritmo de tipo particional basado en grafos. Entre sus ventajas respecto a otros algoritmos de agrupamiento están: la no necesidad de conocer la cantidad de grupos a formar, no impone restricciones para la representación de los objetos ni supedita a estos a una medida de semejanza específica. Por otro lado entre sus desventajas podemos mencionar: la dependencia de los grupos obtenidos respecto al orden de análisis de éstos, no permite adherir o eliminar objetos múltiples durante el proceso y obtiene bastantes grupos con pocos elementos [12].

El algoritmo estrella se representa por medio de un grafo de similaridad de la forma $G = (V, E, w)$ donde los vértices (V) representan los textos y cada arista (E) muestra la similaridad entre dos documentos; finalmente el valor del umbral (w) determina el peso de cada arista. La similaridad entre dos documentos se mide con base en alguna medida de similitud como: la cosenoidal, Euclidiana, Jaccard o Manhathan. Las características sobresalientes de este algoritmo son la posibilidad de tener diferentes *estrellas*.

Algoritmo *k-means*. Este algoritmo es uno de los más utilizados para realizar agrupamiento de datos. Esta técnica tiene como objetivo dividir un conjunto de N elementos en un número predeterminado de grupos k [6]. El algoritmo es sencillo y eficiente, pues permite procesar patrones de forma secuencial. La desventaja del k-means radica en que los primeros patrones determinan la configuración inicial de los grupos y su comportamiento depende enormemente del parámetro k. La idea intuitiva es determinar k centros para cada grupo, luego la distancia entre cada centro determinará el resultado del agrupamiento, por tal razón se recomienda ubicarlos lo más alejados posible entre ellos.

3.6. Evaluación del agrupamiento

La medida de validez interna del agrupamiento, que es la que utilizaremos en este trabajo, consiste en determinar dos aspectos del agrupamiento realizado por un algoritmo dado: por un lado qué tan cohesionados están los grupos entre sí, *i.e.*, se busca que los elementos de un mismo grupo se parezcan más entre ellos que con elementos en otros grupos; y por otro lado determina qué tan separados son los elementos de un grupo con respecto a todos los elementos de otros grupos. Un ejemplo de una medida de evaluación de agrupamiento es el coeficiente de Silhouette.

Coeficiente de Silhouette. El coeficiente de Silhouette muestra qué objetos yacen completamente dentro de un grupo y cuáles están en algún sitio entre grupos. Esta medida tiene un rango de [-1, 1], un valor de 1 indica que el documento está lejos de agrupamientos vecinos, 0 indica que el documento está en o muy cerca de la frontera de decisión entre dos grupos vecinos, y valores negativos indican que el documento podría haber sido mal asignado al grupo.

Tabla 1. Estadísticas de los documentos en el corpus empleado

	Dominio origen	Dominio destino
Num. documentos	1000	1000
Promedio de palabras por docs.	9.8	9.9
Promedio de caracteres por docs.	64.25	65.82
Longitud promedio de palabra	9.5	9.62
Diversidad léxica	1	1.02
Vocabulario promedio por docs.	6.1	6.23

Un promedio de Silhuette cercano a 1 indica que los documentos están agrupados correctamente. Mediante los valores de este coeficiente se puede decidir el número de grupos a formar de un conjunto de datos; sin embargo, esto depende en gran medida del método de agrupamiento utilizado. En los experimentos realizados en este trabajo, la medida de *Silhouette* resulta útil para seleccionar el número de grupos apropiado cuando se utiliza el algoritmo *k-means*; por su parte, en el algoritmo estrella, no es posible modificar el número de grupos generados, utilizaremos entonces el coeficiente para validar la efectividad de los agrupamientos y así comparar resultados entre algoritmos.

4. Experimentos y resultados

4.1. Colección de documentos

Para validar nuestra propuesta realizamos experimentos con los datos proporcionados por el RepLab 2013, el cual está formado por aproximadamente 142,000 tuits, tanto en Inglés como en Español. Este corpus está dividido en cuatro grandes dominios: autos, bancos, universidades y música. Cada dominio contiene diferente número de entidades[6]. Para la recolección de este corpus se realizaron búsquedas en Twitter utilizando el nombre del dominio como parámetro de búsqueda, entre el 1 de Junio de 2012 y el 31 de Diciembre de 2012. Cada una de las 61 entidades consideradas en los cuatro dominios tienen alrededor de 2,200 tuits [1]. Para probar nuestra hipótesis se usaron 1000 tuits de los dominios *autos* y *universidades* (500 por dominio), este conjunto de tuits fue considerado para ajustar los parámetros de los algoritmos de agrupamiento. Posteriormente se usaron otros 1000 tuits de los dominios *bancos* y *música* (500 por dominio) para evaluar el agrupamiento en un enfoque de dominio cruzado.

En la Tabla 1 se muestran algunas estadísticas sobre el conjunto de datos utilizados, el conjunto de documentos del dominio origen corresponden a los dominios del RepLab autos y universidades, mientras que el conjunto de documentos del dominio destino corresponden a los dominios del RepLab: bancos y música. En la tabla se pueden observar estadísticas similares para los dos conjuntos de datos, por lo que la diferencia principal radicará en los temas de los documentos de cada dominio.

[6] Una entidad puede ser una figura pública, empresa, institución o un producto.

Tabla 2. Valor del coeficiente de Silhouette (CS) para agrupamiento del algoritmo k-means para 11 valores distintos de k

k:	10	20	30	40	50	60	70	80	100	150	180
BOW	0.05	0.03	0.02	-0.02	0.04	0.09	0.10	0.05	0.02	0.29	**0.31**
pt_T	0.04	-0.13	-0.03	0.00	0.09	0.06	0.05	0.07	0.17	**0.32**	0.24

Tabla 3. Valor del coeficiente de Silhouette (CS) para agrupamiento del algoritmo estrella aplicado a los dominios destino, bancos y música

	Núm. de grupos	Coeficiente de Silhouette
BOW	36	0.03
pt_T	499	0.16

4.2. Resultados de agrupamiento para el dominio origen

Para comprobar la hipótesis planteada en este trabajo, se realizaron experimentos con dos tipos de representaciones, una representación de alta dimensionalidad obtenida mediante la Bolsa de Palabras (BOW) y una representación de baja dimensionalidad obtenida mediante el método de Punto de Transición (pt_T).

Por lo tanto, el primer experimento consiste en utilizar el algoritmo k-means (con distintos valores de k) con ambas representaciones. Los resultados de este experimento se muestran en la Tabla 2. En la tabla se observa que el mejor agrupamiento para ambas representaciones se obtiene para $k = 150$ y $k = 180$, respectivamente. Otro aspecto a notar en la tabla es que cuando se utilizan la representación de BOW, con 2429 términos, se requieren 180 grupos mientras que cuando se utiliza la representación basada en el Punto de Transición (pt_T), con 674 términos, se requieren 150 grupos para obtener resultados ligeramente mejores. Note para todos los casos se buscan valores del coeficiente de Silhouette cercanos a 1 (vea la subsección 3.6).

Durante los experimentos la variación del valor de k llegó hasta 499, para la cual el valor de Silhouette fue prácticamente 1; sin embargo, para los intereses de esta investigación, esto no fue considerado como un resultado deseable para realizar la validación de dominio cruzado, pues prácticamente está considerando dos documentos por grupo.

Para el segundo experimento, se evaluó el desempeño del algoritmo estrella con las mismas dos representaciones. Los resultados de este experimento se muestran en la Tabla 3. En general el desempeño del algoritmo estrella no es adecuado para este conjunto de datos pues aunque pareciera que el agrupamiento con la representación de Punto de Transición es mejor, el número de grupos que se forman son casi la mitad del número total de documentos, por lo que el agrupamiento es demasiado especializado para este corpus.

Tabla 4. Valor del coeficiente de Silhouette (CS) para agrupamiento del algoritmo k-means para 11 valores distintos de k

k:	10	20	30	40	50	60	70	80	100	150	180
pt_T	0.05	0.07	0.08	0.04	0.02	0.05	0.13	0.02	0.17	0.30	**0.34**

De los dos experimentos anteriores se puede concluir que la mejor configuración resulta de utilizar el algoritmo k-means con $k = 150$ y la representación de baja dimensionalidad, esto es, utilizando el método de reducción de dimensionalidad del Punto de Transición.

4.3. Resultados de agrupamiento para el dominio destino

La mejor configuración de parámetros obtenida en el conjunto de documentos del dominio origen se utilizaron para agrupar el conjunto de documento del dominio destino. Cabe mencionar que el dominio destino está compuesto por tuits de los dominios del RepLab banco y música.

Para realizar este experimento en dominios cruzados se representó cada documento del dominio destino con el vocabulario obtenido con el Punto de Transición en el dominio origen. Es decir, se representaron los documento con 674 términos. Los resultados de esta evaluación se muestran en la Tabla 4.

Como puede verse en los resultados de dominio cruzado, la evaluación del agrupamiento con 180 grupos usando una representación de baja dimensionalidad es incluso mejor que para el dominio origen, *i.e.*, $CS = 0.34$ vs $CS = 0.32$, para el escenario de dominio cruzado y el escenario in-domain, respectivamente.

5. Conclusiones

En este trabajo hemos presentando una metodología para resolver el problema de identificación de tópicos en documentos cortos, específicamente tuits. El problema que da origen a este trabajo proviene de las actividades cotidianas que debe realizar un Analista de Reputación en Línea, entre las cuales se encuentra la identificación de temáticas relevantes a la entidad y/o figura pública de su interés.

Debido a la dinámica con que se genera información en las redes sociales, el pensar en diseñar esquemas de clasificación supervisados se vuelve una tarea inimaginable, pues representaría un proceso muy costoso, mismo que eventualmente se volvería obsoleto al poco tiempo debido a que en este tipo de ambientes (Twitter) las temáticas son muy diversas y constantemente cambiantes.

Así entonces, el trabajo realizado se orientó a tratar de eliminar las limitaciones de los esquemas tradicionales, para lo cual se emplearon estrategias de clasificación no supervisadas. Por otro lado, con la finalidad de construir representaciones robustas a distintos dominios, se utilizó una forma de representación compacta de los documentos. La representación empleada permite eliminar términos muy especializados (dependientes del dominio) así como términos

muy comunes (ruido). Los resultados obtenidos son alentadores pues sugieren la posibilidad de llevar a cabo el agrupamiento de este tipo de textos, en dominios cruzados, con resultados satisfactorios. Durante la realización de los experimentos, fue posible notar que los valores obtenidos del coeficiente de Silhouette para el agrupamiento k-means bajo una representación de punto de transición, comprueba la factibilidad de utilizar este tipo de representaciones en dominios diferentes al que fue empleado para la construcción de la representación inicial.

Como trabajo futuro nos interesa evaluar el desempeño del algoritmo de agrupamiento estrella empleando estrategias diferentes para la definición del umbral de similitud, así como métricas distintas, por ejemplo la la similitud suave de coseno, descrita en [15]. En lo que respecta a los algoritmos de agrupamiento, contrario al método de k-means, el algoritmo estrella no requiere de especificar un número de grupos que se quieren formar, por lo cual creemos que si se logra definir un esquema apropiado para determinar el valor del umbral, el comportamiento de éste será más satisfactorio. Agregado a esto, nos interesa igualmente evaluar la propuesta con una muestra mayor de datos, e incluso incorporando información de los otros años del RepLab.

Agradecimientos. Este trabajo fue parcialmente financiado por el CONACyT a través de las becas 587804 y 588090, y el programa del SNI. También se agradece el apoyo otorgado a través de la Coordinación de la Maestría en Diseño, Información y Comunicación (MADIC) de la UAM Cuajimalpa.

Referencias

1. Amigó, E., Carrillo de Albornoz, J., Chugur, I., Corujo, A., Gonzalo, J., Martín, T., Meij, E., Rijke, M., Spina, D.: Information Access Evaluation. Multilinguality, Multimodality, and Visualization: 4th International Conference of the CLEF Initiative, CLEF 2013, Valencia, Spain, September 23-26, 2013. Proceedings, pp. 333–352. Springer Berlin Heidelberg, Berlin, Heidelberg (2013), http://dx.doi.org/10.1007/978-3-642-40802-1_31
2. Amigó, E., Carrillo-de Albornoz, J., Chugur, I., Corujo, A., Gonzalo, J., Meij, E., Rijke, M., Spina, D.: Information Access Evaluation. Multilinguality, Multimodality, and Interaction: 5th International Conference of the CLEF Initiative, CLEF 2014, Sheffield, UK, September 15-18, 2014. Proceedings, chap. Overview of RepLab 2014: Author Profiling and Reputation Dimensions for Online Reputation Management, pp. 307–322. Springer International Publishing, Cham (2014), http://dx.doi.org/10.1007/978-3-319-11382-1_24
3. Amigó, E., Corujo, A., Gonzalo, J., Meij, E., de Rijke, M.: Overview of RepLab 2012: Evaluating online reputation management systems. (2012)
4. Banerjee, S., Ramanathan, K., Gupta, A.: Clustering short texts using wikipedia. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 787–788. SIGIR '07, ACM, New York, NY, USA (2007), http://doi.acm.org/10.1145/1277741.1277909
5. Booth, A.D.: A "law of occurrences for words of low frequency. Information and Control 10(4), 386 – 393 (1967), http://www.sciencedirect.com/science/article/pii/S001999586790201X

6. Gan, G., Ma, C., Wu, J.: Data clustering: theory, algorithms, and applications, vol. 20. Siam (2007)
7. He, H., Chen, B., Xu, W., Guo, J.: Short text feature extraction and clustering for web topic mining. In: Semantics, Knowledge and Grid, Third International Conference on. pp. 382–385 (Oct 2007)
8. Hu, X., Zhang, X., Lu, C., Park, E.K., Zhou, X.: Exploiting wikipedia as external knowledge for document clustering. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 389–396. KDD '09, ACM, New York, NY, USA (2009), http://doi.acm.org/10.1145/1557019.1557066
9. Mateos Sánchez, M., García-Figuerola Paniagua, C.: Aplicación de técnicas de clustering en la recuperación de información web. (Biblioteconomía y administración cultural ; 205), Gijón (Asturias) : Trea (2009), http://www.mcu.es/ccbae/es/consulta/registro.cmd?id=173815
10. Ni, X., Quan, X., Lu, Z., Wenyin, L., Hua, B.: Short text clustering by finding core terms. Knowl. Inf. Syst. 27(3), 345–365 (Jun 2011), http://dx.doi.org/10.1007/s10115-010-0299-7
11. Porter, M.F.: An algorithm for suffix stripping. Program 14(3), 130–137 (1980)
12. Pérez Suárez, A., Martínez Trinidad, J., Medina Pagola, J., Carrasco Ochoa, A.: Algoritmos dinámicos para el agrupamiento con traslape. Tech. Rep. CCC-10-001, Instituto Nacional de Astrofísica, Óptica y Electrónica (2010)
13. Rosa, K.D., Shah, R., Lin, B., Gershman, A., Frederking, R.: Topical clustering of tweets. Proceedings of the ACM SIGIR: SWSM (2011)
14. Sebastiani, F.: Machine learning in automated text categorization. ACM Comput. Surv. 34(1), 1–47 (Mar 2002), http://doi.acm.org/10.1145/505282.505283
15. Sidorov, G., Gelbukh, A., Gómez-Adorno, H., Pinto, D.: Soft similarity and soft cosine measure: Similarity of features in vector space model. Computación y Sistemas 18(3), 491–504 (2014)
16. Zhuang, F., Luo, P., Yin, P., He, Q., Shi, Z.: Concept learning for cross-domain text classification: A general probabilistic framework. In: Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence. pp. 1960–1966. IJCAI '13, AAAI Press (2013), http://dl.acm.org/citation.cfm?id=2540128.2540409
17. Zipf, G.: Human behaviour and the principle of least-effort. Addison-Wesley, Cambridge, MA (1949), /brokenurl#http://publication.wilsonwong.me/load.php?id=233281783

Modelado de un sistema multi-agente aplicado a la predicción de la personalidad en Twitter

Christian Padilla-Navarro[1,3], Miguel Rebollo-Pedruelo[1], Carlos Lino-Ramírez[2]

[1] Universidad Politécnica de Valencia,
Departamento de Sistemas Informáticos y Computación, Valencia,
España

[2] División de Estudios de Posgrado e Investigación,
Instituto Tecnológico de León, León, Gto.,
México

[3] Universidad Politécnica de Juventino Rosas,
Departamento de Telemática, Santa Cruz de Juventino Rosas, Gto.,
México

jopana2@doctor.upv.es, mrebollo@upv.es, carloslino@itleon.edu.mx

Resumen. En la presente investigación se realiza la simulación del comportamiento de un usuario de Twitter, a través de un sistema multi-agente, obteniendo las características de la personalidad de sus seguidores, aplicando el test de personalidad de los cinco grandes y los clasificadores Naive Bayes y KNN.

Palabras clave: Sistema multi-agente, test de los cinco grandes, Twitter, naive Bayes, KNN.

Multi-agent System Applied to Prediction of Personality on Twitter

Abstract. In this research the simulation of the behavior of a Twitter user is performed through a multi-agent system, obtaining the characteristics of the personality of his followers, applying the personality test of the Big Five and the classifiers Naive Bayes and KNN.

Keywords: Multi-agent system, big five test, Twitter, naive Bayes, KNN.

1. Introducción

Las redes sociales son hoy en día una de las formas de comunicación más importantes que hay en el mundo. En 2005 se estimaba que, sumando todas

las redes sociales existentes, había unos 115 millones de usuarios activos en el mundo [6], mientras que en 2015 se estimaba que tan solo Twitter tenía unos 316 millones de cuentas activas [5].

El vaciado de información que realizamos día con día nos enfrenta a expresar sentimientos, compartir opiniones, conversar con personas distantes, entre otros más, que nos ayudan a formar un "perfil social". La red social Twitter, limitada a 140 caracteres por cada publicación (Tweet), es una de estas formas de expresión.

Durante años la psicología ha estudiado la personalidad del ser humano, y por tanto algunos investigadores han generado diferentes modelos para este fin. Las relaciones han sido descubiertas detrás de la personalidad y el desorden psicológico [13], el desempeño en el trabajo [1] y la satisfacción [9], e incluso algunos sucesos románticos [15].

Debido entonces a la gran cantidad de información vertida en las redes sociales, y a los millones de usuarios que pertenecen a ellas, el análisis de personalidad a través de redes sociales es uno de los objetos de estudio en tendencia. En [8] se aplicó el test de personalidad de los cinco grandes en la predicción de la personalidad de usuarios en Twitter. En [11] se aplicaron redes neuronales y el test de los cinco grandes aplicado a la predicción de la personalidad de usuarios de Twitter.

Simular el comportamiento de uno o varios usuarios en una red a través de agentes o de un sistema multi-agente ha sido un auxiliar importante en diversas investigaciones [18], [17].

Nosotros proponemos simular, a través de un sistema multi-agente, la interacción de un usuario de Twitter con sus seguidores aplicando clasificadores (KNN y Naive Bayes), utilizando las características del test de personalidad de los cinco grandes obtenidas de la API Watson de IBM Bluemix [2].

2. Antecedentes y trabajo relacionado

2.1. Test de personalidad de los Cinco Grandes (Big Five personality test)

Varios grupos de investigadores independientes descubrieron y definieron los cinco grandes factores mediante investigación empírica basada en datos. Ernest Tupes y Raymond Christal aportaron el modelo inicial, basado en el trabajo realizado en el Laboratorio de Personal de las Fuerzas Aéreas de los EE. UU. en la década de 1950 [16]. J. M. Digman propuso su modelo de los cinco factores de personalidad en 1990 [4], y Goldman lo llevó a los niveles más altos de las organizaciones en 1993 [7]. En un test de personalidad, para hacer referencia a los rasgos de los Cinco Grandes, también se puede utilizar el Modelo de los Cinco Grandes [3] y los Factores Globales de personalidad [12]. En la Figura 1 se muestran los cinco factores de personalidad.

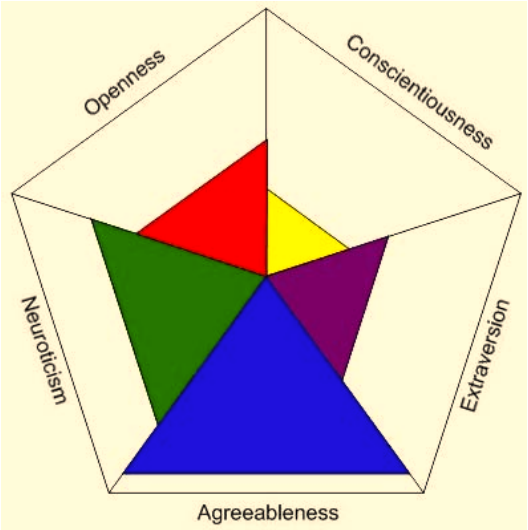

Fig. 1. Test de personalidad de los Cinco Grandes [8]

2.2. Rasgos del test de los Cinco Grandes factores de la personalidad

Apertura a nuevas experiencias. Apertura para apreciar el arte, emoción, aventura, ideas inusuales, imaginación, curiosidad, y variedad de experiencias. Las personas abiertas a las experiencias son intelectualmente curiosos, aprecian el arte, y son sensibles a la belleza. Comparados con personas cerradas tienden a ser más creativos y más concientes de sus sentimientos.

Responsabilidad. Es la tendencia a mantener autodisciplina, actuar responsablemente, y buscar cumplir los objetivos. Este rasgo denota preferencia por lo planeado antes que el comportamiento espontáneo. Influencia el modo en que controlamos, regulamos y dirigimos nuestros impulsos.

Extraversión. Esta caracterizado por emociones positivas y la tendencia a buscar estimulación en la compañía de otros. Un profundo compromiso con el mundo externo. Los extrovertidos disfrutan de la compañía de otros, y son percibidos como personas muy enérgicas. Tienden a ser entusiastas, individuos activos y en busca de emociones. En grupos, les gusta destacar, hablar y llamar la atención.

Amabilidad. Es la tendencia a ser compasivo y coperativo en vez de antipático con los demás. La dimension refleja diferencias personales en general, concernientes a la armonia en sociedad. Individuos amables valoran salir o juntarse con los

demás. Generalmente son considerados, amistosos, generosos, colaboradores y comprometidos con el deseo de los de su entorno.

Neuroticismo. Neuroticismo es la tendencia a experimentar emociones negativas como el enojo, la ansiedad o depresión. También es conocida como inestabilidad emocional.

2.3. Watson IBM BlueMix

El desarrollo en la nube IBM Watson ofrece una gran variedad de servicios para el desarrollo de aplicaciones cognitivas. Cada servicio de Watson proporciona una interfaz de programación de aplicaciones (API) para interactuar con este servicio.

IBM bluemix [2] es la plataforma en la nube en la que se implementan las aplicaciones que se desarrollan en Watson.

Personality Insights. El servicio de percepción de personalidad de Watson de IBM (IBM Watson Personality Insights) proporciona una API que permite a las aplicaciones obtener puntos de vista de los medios sociales, datos de la empresa, u otras comunicaciones digitales. El servicio utiliza un análisis lingüístico para inferir la personalidad y características sociales, incluyendo los cinco grandes, necesidades y valores, a partir del texto. Estos conocimientos ayudan a las empresas a comprender las preferencias de sus clientes y mejorar su satisfacción mediante la previsión de sus necesidades, y además de recomendar acciones futuras. Las empresas pueden utilizar esta información para mejorar la captación de clientes, retención y compromiso, para fortalecer las relaciones con ellos [2].

3. Base de datos

Las características de cada usuario pueden ser vistas en la Tabla 1. Estas se obtuvieron a través del test de personalidad de IBM Bluemix y de la API de Twitter. Para generar dicha base de datos se analizó el perfil de 79 seguidores del usuario @jocripana, que es una cuenta real. El etiquetado de clases se realizó de forma manual por el usuario de la cuenta.

La base de datos utilizada para la clasificación de las características se encuentra formada de la siguiente manera:

3.1. Clasificación de la interacción

La clasificación recae directamente en cuatro clases: Clase 4, el agente tuitero debe seguir al agente seguidor y puede tuitear con él. Clase 3, el agente tuitero no debe seguir al agente seguidor, pero sí puede tuitear con él. Clase 2, el agente tuitero sigue al agente seguidor, pero no tuitea con él. Clase 1, el agente tuitero no sigue ni tuitea con el agente seguidor.

Tabla 1. Características para la clasificación de seguidores de Twitter

No.	Característica	Procedencia	Rango de Valor
1	**Openness** (Apertura)	Watson IBM Bluemix	0 a 100 %
2	Imagination (Imaginación)	Watson IBM Bluemix	0 a 100 %
3	Artistic Interests (Intereses artísticos)	Watson IBM Bluemix	0 a 100 %
4	Emotionality (Emocionalidad)	Watson IBM Bluemix	0 a 100 %
5	Adventurousness (Aventura)	Watson IBM Bluemix	0 a 100 %
6	Intellect (Intelecto)	Watson IBM Bluemix	0 a 100 %
7	Liberalism (Liberalismo)	Watson IBM Bluemix	0 a 100 %
8	**Conscientiousness** (Ser consciente)	Watson IBM Bluemix	0 a 100 %
9	Self-efficacy (Auto-eficacia)	Watson IBM Bluemix	0 a 100 %
10	Dutifulness (Sentido del deber)	Watson IBM Bluemix	0 a 100 %
11	Achievement-striving (Logro-esfuerzo)	Watson IBM Bluemix	0 a 100 %
12	Self-discipline (Auto-disciplina)	Watson IBM Bluemix	0 a 100 %
13	Cautiousness (Cautela)	Watson IBM Bluemix	0 a 100 %
14	**Extraversion** (Extroversión)	Watson IBM Bluemix	0 a 100 %
15	Friendliness (Amabilidad)	Watson IBM Bluemix	0 a 100 %
16	Gregariousness (Gregarismo)	Watson IBM Bluemix	0 a 100 %
17	Assertiveness (Asertividad)	Watson IBM Bluemix	0 a 100 %
18	Activity level (Nivel de actividad)	Watson IBM Bluemix	0 a 100 %
19	Excitement-seeking (Búsqueda de emociones)	Watson IBM Bluemix	0 a 100 %
20	Cheerfulness (Alegría)	Watson IBM Bluemix	0 a 100 %
21	**Agreeableness** (Afabilidad)	Watson IBM Bluemix	0 a 100 %
22	Trust (Confianza)	Watson IBM Bluemix	0 a 100 %
23	Morality (Moralidad)	Watson IBM Bluemix	0 a 100 %
24	Altruism (Altruismo)	Watson IBM Bluemix	0 a 100 %
25	Cooperation (Cooperación)	Watson IBM Bluemix	0 a 100 %
26	Modesty (Modestia)	Watson IBM Bluemix	0 a 100 %
27	Sympathy (Simpatía)	Watson IBM Bluemix	0 a 100 %
28	**Neuroticism** (Neuroticismo)	Watson IBM Bluemix	0 a 100 %
29	Anxiety (Ansiedad)	Watson IBM Bluemix	0 a 100 %
30	Anger (Enfado)	Watson IBM Bluemix	0 a 100 %
31	Depression (Depresión)	Watson IBM Bluemix	0 a 100 %
32	Self-consciousness (Auto-conciencia)	Watson IBM Bluemix	0 a 100 %
33	Immoderation (Falta de moderación)	Watson IBM Bluemix	0 a 100 %
34	Vulnerability (Vulnerabilidad)	Watson IBM Bluemix	0 a 100 %

Tabla 2. Clasificación de la interacción con el seguidor en Twitter

Clase	Lo sigo	Le escribo
Clase 1	X	X
Clase 2	X	✓
Clase 3	✓	X
Clase 4	✓	✓

4. Modelado del sistema multi-agente

4.1. Escenario 1

Para el primer escenario se definieron dos agentes dentro del sistema, el Agente Seguidor y el Agente Tuitero. Dichos agentes tienen atribuciones distintas.

Agentes seguidores. Los agentes seguidores, agentes artificiales, son generados de forma aleatoria. Pueden ser uno o varios, e intentan interactuar con el agente tuitero. Las tareas que tienen a cargo son las siguientes:

- **Enviar un Tweet al Agente tuitero**.- Sucede al ser generados. Es un evento que sucede de forma aleatoria.
- **Dar follow a al Agente tuitero**.- Sucede después de interacturar con el Agente tuitero. Es un evento aleatorio.
- **Responder a un Tweet**.- Sucede en caso de recibir un tweet del Agente tuitero.

Agente tuitero. El agente tuitero, agente artificial, es el centro de la toma de decisiones en el sistema. Este agente cubre con las siguientes tareas:

- **Seguir a un Agente**.- Sucede después de interacturar con el Agente seguidor, y sólo si después de la clasificación está permitida esta tarea.
- **Responder a un Tweet**.- Sucede en caso de recibir un tweet del Agente seguidor, y sólo si después de la clasificación está permitida esta tarea.

Modelado primer escenario. El modelado indica el proceso que seguirá el sistema multi-agente. Cuando llega un nuevo agente seguidor, envía un tweet al agente tuitero. El agente tuitero solicita a la API Watson de IBM Bluemix la extracción de las características del seguidor, a través del envío de un conjunto de tweets obtenido con la API de Twitter, aplicando el test de personalidad de los cinco grandes. Al tener las características las envía al clasificador Naive Bayes, que predice el tipo de interacción que el agente tuitero tendrá con el agente seguidor. Este modelado puede ser visto en la Figura 2.

4.2. Escenario 2

Para el segundo escenario se definieron tres agentes dentro del sistema, el Agente Tuitero, el Agente BlueMix y el Agente Clasificador. Dichos agentes tienen atribuciones distintas.

Agente tuitero. El agente tuitero, agente artificial, es el centro de la toma de decisiones en el sistema. Este agente cubre con las siguientes tareas:

- **Dar follow a un seguidor**.- Sucede después de interacturar con un seguidor, y sólo si después de la clasificación está permitida esta tarea.
- **Responder a un Tweet**.- Sucede en caso de recibir un tweet un seguidor, y sólo si después de la clasificación está permitida esta tarea.

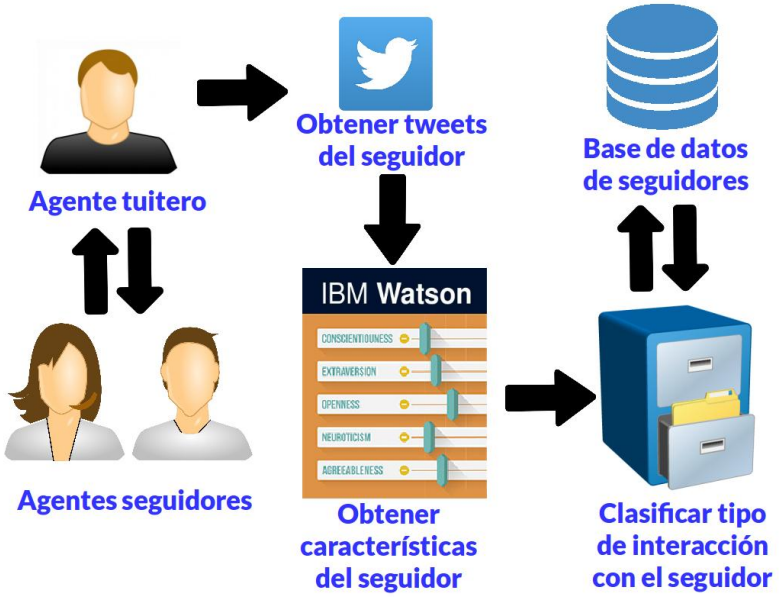

Fig. 2. Modelado del sistema multi-agente para el escenario 1

Agente BlueMix. El agente bluemix, agente artificial, se conecta con la api de Twitter para obtener una cantidad de tweets del seguidor con el que está interactuando. Posterior a esta acción, envía los tweets a la app de IBM BlueMix y extrae las características de los tweets del seguidor aplicando el test de personalidad de los cinco grandes (Watson).

Agente clasificador. Recibe del agente bluemix las características del seguidor con el que está interactuando el agente tuitero. Este agente tiene como tarea clasificar el comportamiento que debe tener el agente tuitero después de haber recibido un tweet de un seguidor o haber sido seguido por el mismo.

Modelado segundo escenario. Para el segundo escenario la interacción realizada es humano-agente. Llega un seguidor seguidor, envía un tweet al agente tuitero. El agente tuitero envía el usuario al agente bluemix. El agente bluemix solicita a la API Watson de IBM Bluemix la extracción de las características del seguidor, a través del envío de un conjunto de tweets obtenido con la API de Twitter, aplicando el test de personalidad de los cinco grandes. Al tener las características, las envía al agente clasificador, que este a su vez las manda al clasificador que realizará la predicción del tipo de interacción que el agente tuitero tendrá con el seguidor. Este modelado puede ser visto en la Figura 3.

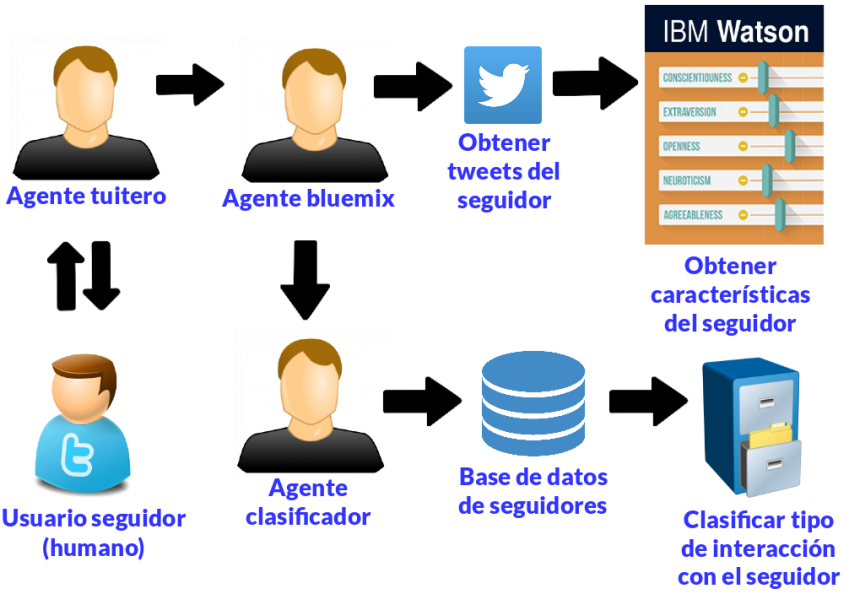

Fig. 3. Modelado del sistema multi-agente para el escenario 2

Tabla 3. Resultados del clasificador KNN

Vecinos	Capas	% de Clasificación
1	3	53.16 %
1	5	53.16 %
1	10	51.89 %
3	3	59.49 %
3	5	55.69 %
3	10	56.96 %
5	3	62.02 %
5	5	59.49 %
5	10	59.49 %
10	3	62.02 %
10	5	63.29 %
10	10	60.75 %

Tabla 4. Resultados del clasificador Naive Bayes

Capas	% de Clasificación
3	46.83 %
5	36.70 %
10	44.38 %

5. Experimentos y resultados

5.1. Clasificación

Para las pruebas previas de clasificación se utilizó el software Weka, y se aplicó en diversos clasificadores, buscando el mayor porcentaje de clasificación para su uso en posterior en la interacción en tiempo real. Las capas y los vecinos fueron propuestos de forma experimental.

6. Conclusiones y trabajo a futuro

Con la metodología propuesta se lograron obtener porcentajes de clasificación superiores al 50 % y en algunos casos incluso superiores al 60 % .

Para la simulación se utilizó SPADE, que es una plataforma libre, de sistemas multi-agente desarrollada en Python y basada en la tecnología de mensajería instantánea XMPP.

Como trabajo a futuro se buscará incluir más características, tales como la cadena de seguidores que tiene el usuario, así como propiedades de la red, el número de triángulos en la red, etc, con el fin de aumentar aún más el porcentaje de clasificación. De igual forma, se buscará aplicar un test similar para Facebook y el desarrollo de una plataforma propia de obtención de características de comportamiento.

Referencias

1. Barrick, M., Mount, M.: The Big Five personality dimensions and job performance: A meta-analysis. Personnel psychology, 44(1):1-–26 (1991)
2. IBM. Información sobre la API Personality Insights de Watson de IBM Bluemix. `https://www.ibm.com/smarterplanet/us/en/ibmwatson/` (Consultado el 07 de Febrero del 2016)
3. Costa, P.T., Jr., McCrae, R.R.: Revised NEO Personality Inventory (NEO-PI-R) and NEO Five-Factor Inventory (NEO-FFI) manual. Odessa, FL: Psychological Assessment Resources (1992)
4. Digman, J.M.: Personality structure: Emergence of the five-factormodel. Annual Review of Psychology, 41, 417–440 (1990)
5. El Mundo. Cuentas activas de Twitter en Septiembre del 2015. `http://www.elmundo.es/tecnologia/2015/09/24/5603bdcbca474105398b4577.html` (Consultado el 07 de Febrero del 2016)
6. Golbeck, J.: Computing and Applying Trust in Web-based Social Networks. PhD thesis, University of Maryland, College Park, MD, USA (Abril 2005)
7. Goldberg, L.R.: The structure of phenotypic personality traits. American Psychologist, 48, 26–34 (1993)
8. Golbeck, J.: Predicting Personality from Twitter. In: IEEE International Conference on Privacy, Security, Risk, and Trust, and IEEE International Conference on Social Computing. DOI: 978-0-7695-4578-3/11 (2011)
9. Judge, T., Higgins,C., Thoresen. C., Barrick, M.: The big five personality traits, general mental ability, and career success across the life span. Personnel psychology, 52(3):621-–652 (1999)

10. Ocean8. Test de personalidad de los cinco grandes. http://ocean.na8.com.ar/blog/2012/11/29/que-es-ocean (Consultado el 07 de Febrero del 2016)
11. Pundlik, M.: A Neural Network Approach to Personality Prediction based on the Big-Five Model. International Journal of Innovative Research in Advanced Engineering (IJIRAE), Chapter 8, Volume 2 (Agosto 2015)
12. Russell, M.T., Karol, D.: 16PF Fifth Edition administrator's manual. Champaign, IL: Institute for Personality & Ability Testing (1994)
13. Saulsman, L., Page, A.: The five-factor model and personality disorder empirical literature: A meta-analytic review* 1. Clinical Psychology Review, 23(8):1055-–1085 (2004)
14. Arana, J. Palanca, J.: SPADE User's Manual. http://www.javierpalanca.com/spade/manual/ (Conultado el 07 de Febrero del 2016)
15. Shaver, P., Brennan, K.: Attachment styles and the 'Big Five' personality traits: Their connections with each other and with romantic relationship outcomes. Personality and Social Psychology Bulletin, 18(5):536 (1992)
16. Tupes, E.C., Christal, R.E.: Recurrent Personality Factors Based on Trait Ratings. Technical Report ASD-TR-61-97, Lackland Air Force Base, TX: Personnel Laboratory, Air Force Systems Command (1961)
17. Vázquez, A., Barrio, I., Vázquez-Salceda, J., Pujol, J.M., Sangüesa, R.: An agent-based Collaboratory. In: ACAI'01, Advanced Course on Artificial Intelligence, Prague (Julio 2001)
18. Watts, D.J., Strogatz, S.H.: Collective dynamics of 'small-world' networks. Nature, 393 (1998)

Método para autocompletar consultas basado en cadenas de Markov y la ley de Zipf

Edgar Moyotl-Hernández, Mónica Macías-Pérez

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

{emoyotl, monica}@fcfm.buap.mx

Resumen. En este trabajo se presenta un algoritmo para autocompletar consultas, el cual genera semiautomáticamente términos que el usuario podría emplear para plantear adecuadamente una consulta y aumentar la efectividad de un Sistema de Recuperación de Información. Con el fin de determinar dichas palabras, se utilizan cadenas de Markov, n-gramas y el punto de transición de Goffman. Este método se aplicó a un corpus general construido con textos de Wikipedia y los resultados obtenidos en los experimentos sugirieron la inclusión de palabras importantes en la formulación de consultas, palabras consideradas relevantes de acuerdo con el modelo de espacio vectorial.

Palabras clave: Expansión de consultas, recuperación de información, cadenas de Markov, n-gramas, ley de Zipf, punto de transición de Goffman.

Method for Autocomplete Queries Based on Markov Chains and Zipf's Law

Abstract. This paper presents an algorithm to autocomplete queries, which semiautomatically generates terms that the user could utilize to properly write a query and increase thereby the effectiveness in an Information Retrieval System. In order to determine these words, Markov chains, n-grams and the Goffman's transition point are used. This method was applied to a general corpus elaborated with texts of Wikipedia and the results obtained in the experiments suggested the inclusion of important words in the query formulation, words considered relevant according to the vector space model.

Keywords: Query expansion, information retrieval, Markov chains, n-grams, Zipf's law, Goffman's transition point.

1. Introducción

La *recuperación de información (RI)* es el proceso por el cual se obtiene un conjunto de documentos cuyo contenido satisface la necesidad de información de

un usuario. Es decir, la recuperación de información intenta resolver el problema de encontrar y ordenar documentos relevantes que satisfagan la necesidad de información de un usuario [1,2]. El ejemplo más popular de esta tarea es el de los sistemas buscadores en Internet tales como Google,[1] Bing[2] o Yahoo[3].

Como se puede notar, el primer paso para llevar a cabo una recuperación consiste en que el usuario exprese su necesidad informativa como una *consulta*. Por lo tanto, el sistema parte de la consulta formulada por el usuario, no de la necesidad de información original, así que una formulación incorrecta o insuficiente (con palabras mal seleccionadas, mal escritas, con faltas de ortografía, etc.) no guiará adecuadamente al sistema durante el proceso de recuperación.

A este respecto, los mayores problemas que enfrenta el *sistema de recuperación de información (SRI)* son, por una parte, la dificultad del usuario para expresar claramente su necesidad en forma de consulta y, por otra, que cuando se describe un mismo concepto, las palabras (o términos) empleadas por el usuario y los autores de los documentos suelen diferir, impidiendo el establecimiento de correspondencias entre las consultas y los documentos [3]. Si bien los usuarios no tienen por qué conocer técnicas de RI, los resultados de su búsqueda mejorarían si se implementan técnicas de *expansión de consultas* (*QE*, por sus siglas en inglés, *Query Expansion*) para lograr que en la respuesta los documentos recuperados sean los documentos relevantes [1].

En este trabajo se presenta un método semiautomático para autocompletar consultas, el cual sugiere al usuario las palabras que podría emplear para especificar adecuadamente una consulta y mejorar su búsqueda, en un SRI que utilice el modelo espacio vectorial para la representación de documentos [4]. Para ello, se usa un procedimiento basado en la frecuencia con que aparecen juntas secuencias de n palabras contiguas, los *n-gramas*. A partir de estas propiedades estadísticas extraídas de un corpus, se construye una *cadena de Markov* para predecir términos de búsqueda que coincidan con el contenido de algún documento, con el fin de aumentar la probabilidad de que se obtengan resultados relevantes. Al mismo tiempo, se explora el uso del *punto de transición de Goffman*, derivado de la *ley de Zipf*, para identificar las palabras con un alto valor semántico en el contenido temático de un texto y la posibilidad de sugerir la inclusión de estos términos en las consultas.

Por otra parte, y a pesar de que existen ya varios trabajos relacionados con la temática, la gran mayoría de la investigación llevada a cabo se centra casi exclusivamente en textos escritos en inglés. Es por ello que los experimentos se realizaron sobre un corpus, no específico con respecto al género, de artículos de Wikipedia escritos en español.

El presente trabajo está organizado de la siguiente manera. En la sección 2, se introducen las características del modelado del lenguaje con cadenas de Markov, se describe el concepto de n-grama y se muestra su uso en la representación de frases. Posteriormente, en la sección 3 se revisan las aplicaciones del punto

[1] https://www.google.com
[2] https://www.bing.com
[3] https://www.yahoo.com

de transición de Goffman en la representación de textos. Luego, en la sección 4 se presenta un resumen de trabajos previos relacionados con el proceso de expansión de consultas y con los algoritmos propuestos, además estos últimos se describen a detalle. A continuación, en la sección 5 se describe el corpus, el proceso de preparación de datos y los experimentos, también se analizan los resultados. Finalmente, en la sección 6 se presentan las conclusiones y el trabajo futuro.

2. Modelado del lenguaje

Cualquier sistema informático que intente tratar el lenguaje natural necesita un modelo que le permita caracterizar y representar la lengua que trata. En los modelos de lenguaje probabilistas, el lenguaje es considerado como una fuente que genera una secuencia de palabras a partir de un conjunto finito de elementos, $V = \{w_i\}$, el vocabulario; y el objetivo es determinar la probabilidad de una secuencia de palabras específica [5], [6]. Esto es útil en diferentes aplicaciones, por ejemplo, corrección ortográfica y gramatical, traducción automática o asistida, reconocimiento de voz y de escritura, predicción de palabras en curso de captura, entre otras.

Como se mencionó anteriormente, el modelo probabilista se encarga de estimar la probabilidad de una frase, para ello, una frase s de longitud L se representa por una secuencia de palabras w_i, es decir, $s = w_1, \ldots, w_L$ o bien $s = w_{1,L}$. Así que, si se interpreta una frase como una sucesión de eventos dependientes, entonces, la probabilidad de cada palabra w_i depende de su historia o contexto $w_1, \ldots, w_{i-1}$, esto es:

$$\begin{aligned} P(s) &= P(w_{1,L}) \\ &= P(w_1)P(w_2|w_1)P(w_3|w_1, w_2)\ldots P(w_L|w_{1,L-1}) \\ &= \prod_{i=1}^{L} P(w_i|w_{1,i-1}). \end{aligned} \tag{1}$$

Considere, por ejemplo, la frase $s =$ *"a b c d e"*. En este caso la probabilidad de s es:

$$\begin{aligned} P(s) &= P(a\ b\ c\ d\ e) \\ &= P(a)P(b|a)P(c|a, b)P(d|a, b, c)P(e|a, b, c, d). \end{aligned}$$

Nótese que el modelo puede predecir la siguiente palabra de una oración a partir de las palabras anteriores. La problemática consiste ahora en estimar la probabilidad $P(w_i|w_{1,i-1})$ para toda palabra del vocabulario y para todo contexto posible, es decir, tomada entre todas las secuencias posibles de palabras de V.

2.1. Cadenas de Markov

Se sabe que al generar una frase puede utilizarse cualquier palabra del conjunto previamente especificado, V, el léxico o vocabulario. Suponga que la frase evoluciona o cambia al agregar palabras a lo largo del tiempo, y sea w_t la última palabra de la frase en el tiempo t. Si se considera que las palabras futuras no están determinadas por las previas (la frase evoluciona de forma no determinista), entonces puede considerarse que w_t es una variable aleatoria para cada valor de t. Esta colección de variables aleatorias es la definición de *proceso estocástico*, y sirve como modelo para representar la evolución aleatoria de una frase a lo largo del tiempo.

Entonces, una determinada frase puede ser interpretada como la realización de un proceso estocástico de tiempo discreto. Si se considera que la próxima palabra (la evolución de este proceso) depende solamente de lo que ya fue escrito (su estado actual), se trata de un proceso markoviano. Por lo tanto, se puede reducir el contexto de la palabra w_i a la palabra más próxima y puede aproximarse de la siguiente forma:

$$P(w_i|w_{1,i-1}) \approx P(w_i|w_{i-1}). \tag{2}$$

Estos tipos de procesos son modelos en donde, suponiendo conocido el estado presente del sistema, los estados anteriores no tienen influencia en los estados futuros del sistema. Esta condición se llama *propiedad de Markov*.

En consecuencia, para aproximar la probabilidad de una oración, simplemente se necesita calcular el producto de las probabilidades de cambiar de un estado (palabra actual) a otro (siguiente palabra), lo que se conoce como *probabilidades de transición*. De modo que, la fórmula para el cálculo de $P(s)$ se obtiene sustituyendo la ecuación 2 en 1 y resulta lo siguiente:

$$P(s) = P(w_{1,L}) \approx \prod_{i=1}^{L} P(w_i|w_{i-1}). \tag{3}$$

Ahora, la probabilidad de la frase $s =$ *"a b c d e"* es:

$$\begin{aligned} P(s) &= P(a\ b\ c\ d\ e) \\ &= P(a)P(b|a)P(c|b)P(d|c)P(e|d)P(e). \end{aligned}$$

Como el conjunto de palabras distintas (el vocabulario) es finito y dado que en una oración la probabilidad de que ocurra una palabra depende de la palabra inmediata anterior, entonces, se puede formar una *cadena de Markov* donde las palabras representen los estados del proceso y la generación de una determinada frase represente la realización del proceso. Lo descrito anteriormente puede representarse gráficamente usando un diagrama como el de la Figura 1.

Los círculos se denominan nodos y representan los estados del proceso, las flechas son los arcos y representan las probabilidades de transición. Note que la suma de los elementos de cada orden es uno. En estas condiciones, es posible,

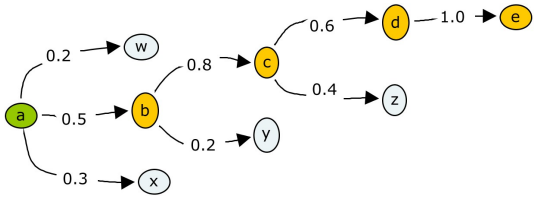

Fig. 1. Cadena de Markov con probabilidades entre palabras

conociendo las probabilidades de transiciones entre estados, calcular la probabilidad de toda cadena particular, en este caso, de una frase específica.

Por consiguiente, si se conocen las probabilidades de las transiciones entre estados, el modelo descrito puede funcionar como generador de consultas. Así por ejemplo, de la cadena representada por la Figura 1, se tiene que dada una primera palabra *"a"* se pueden presentar sugerencias para agregar *"w, b, x"* cada una con una probabilidad distinta y en el siguiente estado *"b"* (en caso de haber sido elegido) nuevamente se podrían presentar más sugerencias de búsqueda *"c, y"*, y así sucesivamente.

2.2. Modelo n-grama

El modelo n-grama es el más empleado y el que mejores resultados ha obtenido dentro del campo de la lingüística computacional. Los *n-gramas* de palabras son secuencias de n palabras consecutivas donde $2 \leq n \leq 7$ para la mayoría de las aplicaciones [5]. Así que, existen 2-gramas (bigramas), 3-gramas (trigramas), 4-gramas, 5-gramas, etc.

Aquí se presenta un ejemplo de n-gramas. De la frase: $s =$ *"a b c d e"* se obtienen los siguientes 2-gramas o bigramas: *a b, b c, c d, d e*. Y los siguientes 3-gramas o trigramas: *a b c, b c d, c d e*. Como se puede observar, el procedimiento es muy sencillo. De manera que, para determinar la probabilidad condicional de que ocurra la palabra w_i después de w_{i-1} se aproxima la probabilidad de un bigrama en particular, esto es:

$$P(w_i|w_{i-1}) \approx \frac{C(w_{i-1}w_i)}{C(w_{i-1})}, \tag{4}$$

en donde $C(w_{i-1}w_i)$ es el número de veces que ocurre el bigrama $w_{i-1}w_i$ y $C(w_{i-1})$ es el número de ocurrencias de la primera palabra del bigrama, w_{i-1}, aproximación presentada en [5], [6]. En suma, el modelo se calcula a partir de la frecuencia de todos los bigramas y de todos los unigramas (es decir, de n-gramas construidos de una sola palabra) en el corpus.

En general, con un modelo n-grama se aproxima la probabilidad de una palabra w_i dadas todas las palabras previas, por la probabilidad de w_i considerando sólo las $n-1$ palabras previas.

3. Ley de Zipf

En general, la mayoría de las palabras frecuentes son también las más cortas y más fáciles de recordar. Por lo que, es evidente que las *palabras funcionales* (también llamadas *palabras vacías* o *stop words*), tales como artículos, pronombres, preposiciones y conjunciones son las más frecuentes en el texto, mientras que las menos frecuentes son palabras que reflejan el estilo y riqueza del vocabulario del autor. Por lo tanto, las palabras que aparecen en la zona media de transición entre las de alta y baja frecuencia de ocurrencia son las que representan al documento [8].

El punto referente a la frecuencia, en torno al cual se encuentran estos términos significativos se llama *punto de transición de Goffman*, puesto que Goffman fue quien introdujo la idea de que las palabras más significativas de un texto se agruparán en una zona donde se encuentran las palabras de alta frecuencia con las de baja frecuencia, es decir, un punto intermedio de transición [9].

3.1. Punto de transición de Goffman

La ley de Zipf y en especial el punto de transición de Goffman han dado buenos resultados en la identificación de palabras clave para la indización y la construcción de tesauros, entre otras aplicaciones [9], [10]. Por ende, en este trabajo se utiliza el punto de transición de Goffman para identificar las palabras clave que representan los textos y se explora la posibilidad de incluir estos términos en las consultas para aumentar la efectividad del proceso de recuperación.

La fórmula para el cálculo del *punto de transición (pt)* es la siguiente:

$$pt = \sqrt{W}, \tag{5}$$

donde W es el número de total de palabras diferentes en el corpus, el tamaño del vocabulario V. La derivación y formulación matemática de esta ecuación puede ser consultada en el trabajo de [11] y [12].

Se puede observar que, cuando la frecuencia de una palabra es idéntica al pt su distancia a él es cero produciendo un valor de cercanía máximo. Por el contrario, si la palabra se encuentra alejada del pt, entonces su distancia aumentará. La medida que calcula esos valores para cada palabra a ambos lados del punto de transición de Goffman, la *distancia inversa al punto de transición (dipt)*, se define en el trabajo de [13] y su fórmula es:

$$dipt_i = \frac{1}{|pt - f_i| + 1}, \tag{6}$$

en donde pt se calcula de acuerdo con la ecuación 5 y f_i es la frecuencia de la palabra w_i en el corpus, es decir, el número de veces que la palabra ocurre en la colección dada. La posibilidad de dividir entre cero está prevista, por eso usa "+1". Con esta medida, los términos más cercanos al pt son aquellos que obtienen los valores más altos. Por supuesto, pt se puede calcular para un sólo texto y $dipt$ se puede calcular para distintas características de los textos incluyendo los n-gramas.

4. Expansión de consultas

En un SRI se suelen utilizar diferentes métodos para optimizar el proceso de búsqueda de información, uno de ellos es la expansión de las consultas ingresadas por el usuario. En general, se trata de un proceso por el que se toma la consulta original ingresada por el usuario y se le amplía con otros términos equivalentes o más adecuados para expresar un concepto [1], [14]. Existen numerosos trabajos sobre QE, en [3] se presenta una revisión de un gran número de métodos recientes que utilizan diversas fuentes de información[4] y emplean diferentes técnicas.

4.1. Trabajos relacionados

Según la literatura revisada, el problema en cualquier tipo de expansión de consultas es cómo definir cuáles términos están relacionados con los de la consulta. Algunos métodos utilizan cadenas de Markov para seleccionar dichos términos y las probabilidades de transición (es decir, las probabilidades de pasar de una palabra a otra) se estiman, en [15] mediante los tipos de relación semántica (polisemia, antonimia, sinonimia, etc.) entre las palabras, en [16] con la combinación de múltiples fuentes de conocimiento sobre sus relaciones y en [17] mediante las relaciones entre los significados de las palabras, todos obtienen excelentes resultados.

Por otra parte, la expansión de consultas puede ser desarrollada manual, automática o semiautomáticamente (interactivamente). En una expansión de consulta interactiva, el sistema sugiere términos y los presenta al usuario, así el usuario es quien toma la decisión final sobre la importancia relativa y la utilidad de un término [14]. Uno de los sistemas más conocidos de este tipo es la función autocompletar de Google,[5] la cual ofrece posibles términos de búsqueda para completar una consulta mientras el usuario está escribiéndola. Las sugerencias de la función autocompletar se generan automáticamente mediante un algoritmo que se basa en una serie de factores, incluida la frecuencia con la que otros usuarios buscaron una palabra y el contenido de las páginas web. Aunque el algoritmo está diseñado para reflejar la variedad de información disponible es posible que no muestre sugerencias para una palabra o un tema en particular.

En este trabajo se presenta un nuevo algoritmo y una variación del mismo para realizar expansión de consultas de forma semiautomática, requiriendo una interacción mínima del usuario. Estos algoritmos se basan en la probabilidad de que dos palabras aparezcan juntas y en la importancia relativa de esas palabras en una colección de documentos. A diferencia de los métodos descritos anteriormente toda la información se obtiene directamente del corpus, sin usar recursos lingüísticos externos.

[4] Los recursos lingüísticos más utilizados en la QE son diccionarios, tesauros y ontologías.

[5] https://support.google.com/websearch

4.2. Métodos propuestos

Las técnicas propuestas están basadas en la frecuencia de n-gramas en la colección de textos, esto garantiza que los usuarios utilicen las mismas palabras que aparecen en los documentos:

1. El método probabilístico combina la idea de la probabilidad condicionada, los n-gramas, con la noción de sucesos encadenados. La probabilidad de generar una determinada frase con este modelo es simplemente el producto de las probabilidades de los bigramas que la conforman. Esta primera propuesta establece un sistema de pesos en función de la probabilidad de cada n-grama en el corpus, como se muestra en la ecuación 3.
2. El método derivado de la ley de Zipf se basa en la idea de que las palabras más significativas de un texto se agruparán en una zona donde se encuentran las palabras de alta frecuencia con las de baja frecuencia, es decir, el punto de transición de Goffman. En este modelo, el peso de la frase se obtiene en función de la probabilidad de cada n-grama y de la distancia inversa de éste al punto de transición (ver ecuación 6).

En consecuencia, los métodos propuestos permiten ordenar las sugerencias con base en los valores de los pesos obtenidos.

4.3. Algoritmo

El algoritmo consiste en guiar al usuario para expandir el término inicial por uno más específico; permitir seleccionar y agregar términos relacionados con los de la consulta con el fin de precisar los documentos a recuperar; representar en forma adecuada un concepto de interés para el usuario. Dicho algoritmo se explica a continuación:

1. Generar y dividir en archivos cada uno de los tipos de n-gramas (unigramas, bigramas, trigramas, etc.) y contabilizar el número de veces que ocurren en el corpus. En los experimentos, el tamaño máximo de los n-gramas utilizados fue de 5.
2. Obtener el primer término de la consulta especificado por el usuario, el cual actúa como término inicial del n-grama.
3. Buscar en el archivo de $(n+1)$-gramas todas las combinaciones de palabras que comiencen con el n-grama y ofrecerlas al usuario como sugerencias. Para que el usuario pueda evaluar cuál es la mejor elección entre esas sugerencias se ordenan de acuerdo con los pesos de la métrica empleada.
4. Obtener el siguiente término de la consulta elegido por el usuario. Si el término está en la lista de sugerencias se agrega al n-grama, en caso contrario se convierte en término inicial de un nuevo n-grama.
5. Repetir los pasos 3 y 4 hasta que se obtengan consultas con los términos deseados o no se encuentren más sugerencias.

5. Experimentos

Ayudar al usuario a buscar es un desafío interesante que puede realizarse antes o durante el proceso de recuperación de información. Este trabajo presenta un estudio experimental sobre un corpus de artículos de Wikipedia escritos en español. El objetivo principal consiste en medir el funcionamiento de las técnicas diseñadas para sugerir términos de búsqueda con datos reales.

5.1. Medida de relevancia

En ambas técnicas se utilizó el valor promedio del parámetro $tfidf$ para evaluar los términos sugeridos, es decir, se obtuvo la suma de los pesos $tfidf_{ij}$ del término t_i en el documento d_j y se dividió por el número de documentos de la colección donde aparece dicho término, la frecuencia de documentos df_i (en inglés, *document frequency*). Los pesos de $tfidf_{ij}$ se normalizaron mediante la *normalización de coseno*, así los pesos de cada uno de los términos oscilan entre cero y uno, puesto que este proceso crea vectores unitarios. Intuitivamente, tf_{ij} mide la importancia relativa de un término en un documento, mientras que idf_i mide la importancia global de un término en todo el conjunto de documentos. El objetivo de tal esquema de pesado es mejorar la discriminación entre documentos y mejorar la efectividad en tareas de recuperación de información [4].

5.2. Wikicorpus

Para la realización de los experimentos se utilizó un corpus de textos de Wikipedia[6]. El Wikicorpus[7] es un corpus trilingüe (catalán, español e inglés) que contiene gran parte de Wikipedia del año 2006 y en su versión 1.0 contiene más de 750 millones de palabras [18]. De los tres corpora sólo se experimentó con el corpus en español; dicho corpus también integra El Corpus del Español Actual (*CEA*)[8].

Para la evaluación se aplicó la técnica de validación cruzada con 10 diferentes partes del corpus (10 *fold-cross validation*). Cada volumen de prueba tiene 25 904 documentos y un tamaño promedio de 45 MB. En la Tabla 1 se resumen estadísticas acerca de los n-gramas en estos volúmenes, el número de n-gramas promedio es obtenido después de eliminar palabras vacías.

El corpus en crudo no asigna categorías a los documentos, así que la experimentación se realizó sobre texto plano. La separación de palabras se llevó a cabo empleando los signos de puntuación, espacios en blanco o combinaciones de ellos como separadores. Una vez obtenidos los términos, se hizo la conversión de todos los caracteres a minúscula. Finalmente, se aplicó la operación de eliminación de palabras vacías.

[6] https://www.wikipedia.org
[7] http://www.cs.upc.edu/ nlp/wikicorpus
[8] http://spanishfn.org/tools/cea/spanish

Tabla 1. Estadísticas de los n-gramas en los volúmenes

	Únicos	Totales	Frec. máxima
1-gramas	263 616	4 111 204	12 426
2-gramas	1 590 716	2 273 378	4 338
3-gramas	1 156 542	1 242 148	425
4-gramas	668 754	687 443	336
5-gramas	373 407	379 219	231

5.3. Análisis de resultados

A continuación se presentan resultados de la aplicación de los métodos propuestos a palabras de distinto nivel de frecuencia en el corpus. En la Tabla 2 se muestran las 15 palabras utilizadas en los experimentos, palabras con frecuencia alta, mediana y baja. La primera columna es la palabra, la segunda es su peso promedio (ver Sección 5.1), la tercera es el valor de su frecuencia de ocurrencia y la cuarta es la frecuencia de documentos, todos son valores promedio en los 10 volúmenes.

Tabla 2. Palabras utilizadas en los experimentos

Palabra	$tfidf$	f_i	df_i	Palabra	$tfidf$	f_i	df_i	Palabra	$tfidf$	f_i	df_i
ciudad	0.039	12 426	4 671	tormentas	0.074	100	57	cosquillas	0.095	3	3
historia	0.025	8 747	5 190	nervios	0.077	100	66	microbio	0.122	3	2
guerra	0.042	6 739	2 643	glaciar	0.129	99	42	tesauro	0.135	3	2
mundo	0.034	5 145	2 825	robots	0.089	98	41	peculado	0.104	2	2
gobierno	0.046	4 892	1 978	guerrilla	0.077	97	60	estornudo	0.078	2	2

Puesto que la probabilidad de ocurrencia de una palabra se calcula dividiendo el número de veces que aparece una palabra entre el número total de palabras, se deduce que si la palabra es muy frecuente tiene mayor probabilidad de ocurrir y la cantidad de documentos a recuperar es enorme, mientras que, si la palabra no es frecuente se recuperarán pocos documentos. Por otro lado, la medida $tfidf$ indica qué tan relevante es la palabra dentro de un documento y en los documentos de la colección, por tanto, una palabra será importante si no es muy frecuente y aparece en pocos documentos (ver Tabla 2).

Para la evaluación de los métodos propuestos, el número de sugerencias se estableció a 10 o menos (ya que todas las sugerencias posibles pueden ser demasiadas). Como se explicó en la sección 4.2, para el enfoque basado en la ley de Zipf, primero se obtiene un cierto número de los n-gramas más probables (en los experimentos realizados se uso la cuarta parte del total de sugerencias

obtenidas) y luego, se presentan al usuario únicamente los más cercanos al punto de transición, en este caso al menos 10. Por ello, los términos sugeridos por ambos métodos podrían ser distintos.

Como ejemplo de la implementación del algoritmo (en uno de los volúmenes de prueba), en las Tablas 3, 4 y 5 se muestran las sugerencias obtenidas y el número de documentos a recuperar para las consultas *"guerra"* , *"nervios"* y *"cosquillas"*, cuando se elige la primera opción. Cabe mencionar que donde la sugerencia está vacía (es nula) es porque el algoritmo no encontró resultados o se alcanzó el número máximo de sugerencias, 5 en este caso.

Tabla 3. Sugerencias obtenidas para *"guerra"*

Método probabilístico		Método con la ley de Zipf	
Consulta	Sugerencia/Docs.	Consulta	Sugerencia/Docs.
guerra	mundial/804	guerra	independencia/189
	civil/558		civil/558
	independencia/189		vietnam/48
	estados/21		carlista/30
	santa/14		treinta/31
	muerte/5		marina/23
	ciudad/2		franco/31
	brasil/13		anglo/23
	alemania/16		malvinas/19
	paz/15		troya/19
guerra **mundial**	alemania/1	guerra **independencia**	estados/19
	grupo/1		venezuela/8
	ciudad/1		grecia/6
	movimiento/1		turca/4
	francia/2		alto/3
guerra **mundial**	**alemania** nazi/1	guerra **independencia**	**estados** unidos/19
guerra **mundial**	**alemania nazi**	guerra **independencia**	**estados unidos** aliada/1
		guerra **independencia**	**estados unidos aliada**

Nótese que aunque las frecuencias de los n-gramas son mucho más bajas que las frecuencias de las palabras simples, su uso aumenta la cantidad de documentos relevantes obtenidos para la consulta dada, por lo que disminuye la cantidad de documentos recuperados. Esta característica es útil en colecciones de documentos más grandes como la Web, en donde la consulta necesita ser más precisa para obtener más páginas relevantes.

Los resultados globales de los experimentos con las diferentes consultas de prueba se reportan en la Tabla 6. En ella se muestran las palabras iniciales de las consultas, la ponderación promedio de los términos sugeridos (de acuerdo con la función $tfidf$) y el promedio de documentos a recuperar con esas sugerencias. Estos resultados muestran que, en general, las dos técnicas sugieren términos

Tabla 4. Sugerencias obtenidas para *"nervios"*

Método probabilístico		Método con la ley de Zipf	
Consulta	Sugerencia/Docs.	Consulta	Sugerencia/Docs.
nervios	papel/1	nervios	craneales/5
	fuerza/1		longitudinales/1
	manos/1		laterales/2
	sonido/1		cruzados/2
	hojas/1		dibujan/2
	bastante/1		radiales/1
	capilla/1		bastante/1
	aparecen/1		encargados/1
	laterales/2		apoyan/1
	arco/1		perceptivos/1
nervios **papel**	operador/1	nervios **craneales**	opuesto/1
			contralateral/1
nervios **papel operador** arena/1		nervios **craneales opuesto**	
nervios **papel operador arena**			

Tabla 5. Sugerencias obtenidas para *"cosquillas"*

Método probabilístico		Método con la ley de Zipf	
Consulta	Sugerencia/Docs.	Consulta	Sugerencia/Docs.
cosquillas	lengua/1	cosquillas	lengua/1
	plantas/1		plantas/1
cosquillas **lengua**		cosquillas **lengua**	

relevantes a las consultas, si se considera que un término es relevante cuando recupera documentos que contienen los términos de la consulta.

Al comparar los datos mostrados en la Tabla 6 se observa que la técnica derivada de la ley de Zipf obtuvo mejores resultados al sugerir términos más importantes desde el punto de vista del modelo vectorial. A partir de este hecho se deduce que dicha técnica complementa a la técnica probabilística al agregar términos importantes y también comunes, términos de frecuencia media. Esto demuestra que el punto de transición de Goffman se puede utilizar para identificar palabras o n-gramas con un alto valor semántico en el contenido temático de textos.

Finalmente, otro punto interesante a destacar es que únicamente en consultas con palabras de frecuencia extremadamente baja, las dos técnicas van a sugerir las mismas palabras y recuperar los mismos documentos.

6. Conclusiones

Este trabajo explora una aplicación de las cadenas de Markov y la ley de Zipf para autocompletar consultas, es decir, generar secuencias de términos

Tabla 6. $tfidf$ promedio de los términos sugeridos/Documentos promedio a recuperar

	Método probabilístico			Método con la ley de Zipf		
Consulta	1er. Sug.	2da. Sug.	3ra. Sug.	1er. Sug.	2da. Sug.	3ra. Sug.
ciudad	0.050/63	0.056/6	0.083/1	0.059/80	0.072/1	0.072/1
historia	0.048/32	0.075/1	0.067/1	0.054/37	0.070/2	0.073/1
guerra	0.050/154	0.049/2	0.066/2	0.082/90	0.077/6	0.069/5
mundo	0.040/12	0.059/1	0.085/1	0.058/17	0.061/1	0.070/1
gobierno	0.047/38	0.072/7	0.067/1	0.055/41	0.073/2	0.047/4
tormentas	0.056/2	0.061/1	0.056/1	0.068/2	0.065/1	0.061/1
nervios	0.053/1	0.071/1	0.065/1	0.076/1	0.073/1	0.087/1
glaciar	0.049/1	0.069/1	0.061/1	0.123/1	0.083/1	0.057/1
robots	0.048/1	0.080/1	0.063/1	0.073/1	0.066/1	0.052/1
guerrilla	0.054/1	0.144/1	0.057/1	0.092/2	0.065/1	0.085/1
cosquillas	0.064/1	0.048/1	0.081/1	0.064/1	0.056/1	0.081/1
microbio	0.061/1	0.063/1	0.163/1	0.061/1	0.055/1	0.172/1
tesauro	0.069/1	0.066/1	0.074/1	0.069/1	0.053/1	0.131/1
peculado	0.060/1	0.091/1	0.103/1	0.060/1	0.086/1	0.086/1
estornudo	0.056/1	0.057/1	0.058/1	0.056/1	0.210/1	0.063/1

potencialmente útiles para obtener documentos relevantes a una necesidad de información. Las ventajas de emplear este enfoque para la expansión de consultas son, por un lado, su capacidad para aumentar la efectividad en la recuperación de información y, por otro, su sencillez y facilidad para implementarlo.

Los resultados obtenidos, sobre un corpus de textos de Wikipedia en español, demuestran que el método propuesto contribuye a resolver el problema que enfrentan los sistemas de recuperación de información, cuando el usuario parte de una formulación incorrecta de la consulta. La precisión de los sistemas de recuperación de información depende en gran medida de los términos que se encuentran en la consulta, es por ello que, intentar mejorar la consulta puede aumentar la cantidad y calidad de los documentos recuperados para satisfacer la necesidad de información dada por el usuario.

Como trabajo futuro, se planea combinar el método propuesto con los recursos lingüísticos adecuados y evaluar el algoritmo con otras colecciones de documentos y con otros idiomas.

Referencias

1. Baeza, R., Ribeiro, B.: Modern information retrieval, vol. 463. ACM Press, New York (1999)
2. Tolosa, G., Bordignon, F.: Introducción a la Recuperación de Información. Universidad Nacional de Luján, Argentina (2007)
3. Carpineto, C., Romano, G.: A survey of automatic query expansion in information retrieval. ACM Computing Surveys (CSUR), 44(1), article 1 (2012)
4. Salton, G., Wong, A., Yang, C.: A Vector Space Model for Automatic Indexing. Communications of the ACM, vol. 18(11), pp. 613–620 (1975)

5. Moreno, A.: Lingüística Computacional. Editorial Sintesis, Madrid (1998)
6. Manning, C., Schütze, H.: Foundations of Statistical Natural Language Processing. MIT press, Cambridge (1999)
7. Zipf, G. K.: Human Behavior and the Principle of Last-Effort. Addison-Wesley, Cambridge (1949)
8. Luhn, H. P.: A statistical approach to mechanized encoding and searching of literary information. IBM Journal of Research and Development, vol. 1, pp. 309–317 (1957)
9. Urbizagástegui, R., Restrepo, C.: La ley de Zipf y el punto de transición de Goffman en la indización automática. Investigación Bibliotecológica, vol. 25(54), pp. 71–92 (2011)
10. Velasco, M., Díaz, I., Lloréns, J., Amescua, A., Martínez, V.: Algoritmo de filtrado multi-término para la obtención de relaciones jerárquicas en la construcción automática de un tesauro. Revista Española de Documentación Científica, vol. 22(1), pp. 34–49 (1999)
11. Booth, A.: A Law of Ocurrences for Words of Low Frequency. Information and Control, vol. 10(4), pp. 383–396 (1967)
12. Sun, Q., Shaw, D., Davis, C. H.: A model for estimating the occurrence of same-frequency word and the boundary between high-and low-frequency words in text. Journal of the Association for Information Science and Technology, vol. 50(3), pp. 280–286 (1999)
13. Moyotl, E., Jiménez, H.: Experiments in Text Categorization using Term Selection by Distance to Transition Point. Research on Computing Science, vol. 10, pp. 139–146 (2004)
14. Deco, C., Bender, C., Saer, J., Chiari, M.: Expansión de consultas utilizando recursos lingüísticos para mejorar la recuperación de información en la web. (2005)
15. Cao, G., Nie, J. Y., Bai, J.: Using Markov chains to exploit word relationships in information retrieval. In: Large Scale Semantic Access to Content (Text, Image, Video, and Sound). Le Centre de Hautes Etudes Internationales D'informatique Documentaire, pp. 388–402 (2007)
16. Collins-Thompson, K., Callan, J.: Query expansion using random walk models. In: Proceedings of the 14th ACM international conference on Information and knowledge management, pp. 704–711 (2005)
17. Gan, L., Wang, S., Wang, M., Xie, Z., Zhang, L., Shu, Z.: Query expansion based on concept clique for Markov network information retrieval model. In: Fuzzy Systems and Knowledge Discovery, 2008. FSKD'08. Fifth International Conference on, vol. 5, pp. 29–33. IEEE (2008)
18. Reese, S., Boleda, G. , Cuadros, M., Padró, L., Rigau, G.: Wikicorpus: A Word-Sense Disambiguated Multilingual Wikipedia Corpus. In: 7th Language Resources and Evaluation Conference, LREC. La Valleta, Malta (2010)

Análisis de sentimientos basado en aspectos: un modelo para identificar la polaridad de críticas de usuario

Miguel Angel Rosales Quiroga, Darnes Vilariño Ayala, David Pinto, Mireya Tovar, Beatriz Beltrán

Benemérita Universidad Autónoma de Puebla,
Faculty of Computer Science, Puebla,
México

miguelrosales@gmail.com,
{dvilarinoayala,davideduardopinto,mireyatovar,bbeltranmtz}@gmail.com
http://www.lke.buap.mx/

Resumen. Con el crecimiento de los usuarios de internet ha aumentado la cantidad de datos generados en la red por lo que se hace importante desarrollar modelos que permitan obtener información importante a partir de dichos datos. La propuesta presentada en este artículo pretende detectar la polaridad de enunciados, párrafos o fragmentos de texto, mencionadas en reseñas de usuarios. El objetivo es identificar los aspectos y el sentimiento expresado para cada aspecto. Se plantea la creación de un modelo no supervisado para la identificación de las características léxicas sintácticas para la detección de aspectos y la clasificación del sentimiento expresado en las reseñas en tres categorías: positivo, negativo y neutro. Se analizaron características como el etiquetado gramatical de las palabras, la similitud semántica entre palabras, la co-ocurrencia de palabras en un conjunto de documentos.

Palabras clave: Análisis de sentimientos, análisis basado en aspectos, polaridad.

Aspect Based Sentiment Analysis: A Model for Identification of User Reviews Polarity

Abstract. The continued growth of users has generated a huge number of data in Internet, thus leading to the importance of developing new models for obtaining relevant information from this data. In this paper we aim to detect the polarity of user reviews. The purpose of this paper is to identify aspects and sentiments expressed for each aspect. We propose an unsupervised model for the identification of lexical and syntactic features for the detection of aspects and the classification of the sentiment expressed in user reviews into three features: positive,

negative and neutral. Some features as the following ones are employed: part-of-speech tags, semantic similarity among words and co-ocurrence.

Keywords: Sentiment analysis, aspect-based analysis, polarity.

1. Introducción

El Análisis de Sentimientos es una tarea de clasificación de textos dentro del área del Procesamiento del Lenguaje Natural, su objetivo es dado una opinión de usuario poder detectar la polaridad de ésta, ya sea positiva, negativa o neutra. El conocer la opinión que una persona tiene hacia un producto o servicio, es de gran ayuda para la toma de decisiones, ya que permite a otros posibles consumidores detectar la calidad del producto o servicio evaluado para utilizarlo. En este artículo se plantea un modelo no supervisado para resolver el problema del análisis de opiniones basado en aspectos. Para la detección de aspectos y la polaridad de las opiniones se realizan el análisis léxico de las reseñas, así como también la semántica de las mismas. Se trabajó con un conjunto de datos de entrenamiento conformado por reseñas sobre Restaurantes y Laptops, éstas reseñas fueron proporcionadas en el marco del Semeval 2016.

Uno de los primeras investigaciones sobre el análisis de sentimientos fue presentada por Pang y Lee [11]. Publicaron su trabajo sobre la clasificación de documentos en base al sentimiento expresado en éstos. Analizaron reseñas sobre películas, encontraron que las técnicas de Aprendizaje Automático mejoran el rendimiento de las líneas base generadas por los expertos humanos. Emplearon tres algoritmos de Aprendizaje Automático: Naive Bayes (NB), Máxima Entropía (ME) y Máquinas de Soporte Vectorial (SVM).

Uno de los primeros trabajos que introdujeron el término de Análisis de Sentimientos fue el presentado por Nasukawa y Yi [10]. En esta publicación definen esta tarea como encontrar expresiones de sentimientos para un sujeto dado y determinar la polaridad de los mismos. En las investigaciones anteriores a ésta, se realizaba el análisis de la polaridad general de un documento, sin embargo en este enfoque se trata de identificar la opinión de cada sujeto mencionado en el texto.

Minqing Hu y Bing Liu [4], expusieron una propuesta para minar y resumir reseñas de consumidores. Los objetivos de este trabajo fueron encontrar las características a las cuales se hacían referencia en las críticas, identificar los enunciados que expresaban opiniones y polaridad sobre las mismas y resumir los resultados. Se propuso la creación de una pequeña lista de adjetivos "semilla" etiquetados manualmente dependiendo si expresan sentimiento positivo o negativo. Posteriormente esta lista es aumentada usando WordNet [8]. Para la detección de los aspectos se emplearon características de etiquetado de las partes del enunciado (Part of speech tagging). Se identifican las características frecuentes, aunque solo se analizan las que se presentan de manera explícita en los enunciados. A continuación, se realiza una extracción de palabras que expresan opinión, se tiene preferencia por los adjetivos cercanos a los aspectos

para tener enunciados de opinión. La identificación de la orientación de estas palabras se realiza mediante un análisis de sinónimos y antónimos.

Una de las propuestas más interesantes presentadas en el SemEval 2014 y que además ha obtenido los mejores resultados en esta tarea es la presentada por Kiritchenko, Zhu, Cherry y Mohammad [5], presentan técnicas como la creación de diccionarios para la detección de sentimientos, estos creados automáticamente utilizando fórmulas que analizan la información mutua [6]. Las palabras de negación son analizadas en un contexto diferente y se crearon diccionarios para estos casos. Adicional a estos se implementaron diccionarios sobre el dominio de laptops y restaurantes.

Otro trabajo que es importante destacar para este Foro de Competición es el desarrollado por Pavel Blinov y Eugeny Kotelnikov [3], que proponen un método para el Análisis de Sentimientos basado en Aspectos para un conjunto de opiniones sobre laptops y restaurantes. El método propuesto para la extracción de aspectos consiste en dos pasos: la selección de candidatos y la extracción de términos.

En la investigación desarrollada por Schouten, Frasincar y De Jong [13], presentan un enfoque basado en co-ocurrencias para la detección de categorías y uno basado en diccionarios de sentimientos para la clasificación. En particular los datos utilizados son los presentados en el Semeval 2016. A continuación se disctute me metodología propuesta.

2. Metodología

Para darle solución a este problema se propone un modelo compuesto por tres fases:

1. Fase de Pre-procesamiento.
2. Fase de Identificación de Aspectos.
3. Fase de Identificación de Polaridad.

Fase de preprocesamiento Durante esta fase se realiza el análisis de los datos de entrenamiento proporcionados por el SemEval para la tarea, además de reunir los datos y generar los archivos de entrada necesarios para el modelo propuesto. Los datos de entrenamiento se encuentran en formato XML. Estos datos contienen el conjunto de reseñas con las categorías correctas, el objeto al cual se hace referencia en la reseña y la polaridad expresada hacia éste. Para el dominio de Laptops sólo se proporciona la categoría correcta identificada en la reseña.

Del conjunto de datos de entrenamiento se obtienen varios elementos de entrada. Primero, se genera un diccionario con los aspectos que se encontraron en esas reseñas. Después, cada reseña del conjunto de datos de entrenamiento es tratada mediante la herramienta Clips Pattern[1] para obtener su etiqueta

[1] http://www.clips.ua.ac.be/

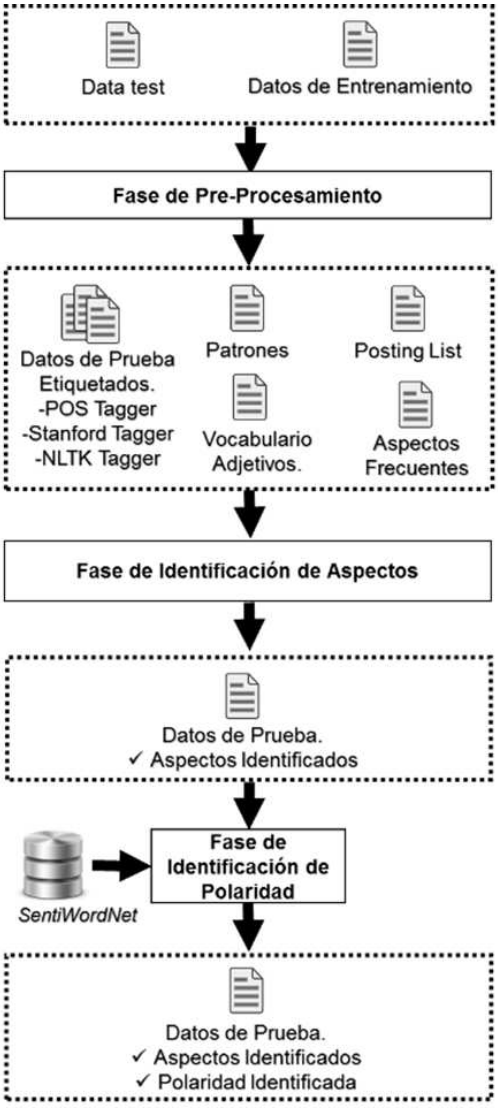

Fig. 1. Metodología basada en tres fases

POS (Part of Speech). Se extraen los patrones de cada aspecto y para cada aspecto se obtienen sus elementos gramaticales adyacentes. Esto para generar un diccionario con los patrones que cumple un aspecto, además de conocer que elementos gramaticales se encuentran generalmente junto a éste.

Una vez que se tienen el conjunto de aspectos de los datos de entrenamiento se genera un Crawler con estos términos para la generación de dos corpus con datos específicos relacionados a Laptops y Restaurantes. Este consiste en para cada

aspecto encontrar documentos relacionados con el mismo en artículos de internet. Es decir, para cada aspecto son extraídos un conjunto de párrafos relacionados. Para su implementación se utilizaron las bibliotecas de BeautifulSoup [9], que permite acceder a páginas web y obtener su contenido.

A continuación, teniendo estos corpus específicos se generó un Posting List para cada corpus. Esto consiste que para cada palabra se genera una lista indexada con el identificador del párrafo que la contiene. Por ejemplo, para el aspecto "Food" se genera una lista con el índice de los párrafos en los que aparece. Con esto, se generaron dos Posting List con 554,356 y 463,924 palabras para Restaurantes y Laptops respectivamente. Finalmente se genera un diccionario con los adjetivos, adverbios y verbos incluidos en los datos de entrenamiento. En este caso con su relación con la polaridad identificada. Como entrada para la siguiente fase, también se genera el etiquetado gramatical del conjunto de datos de prueba (Data test) con los tres etiquetadores propuestos (Stanford Parser [7], Clips Pattern y NLTK [2]).

Fase de identificación de aspectos En esta fase se realiza la detección de los aspectos mencionados en cada reseña. Primero, para cada crítica, se extraen los sustantivos o secuencia de sustantivos y son agregados a un diccionario de candidatos a aspectos indicado con un peso que cumple con esta característica. Posteriormente, bajo la hipótesis de que dos palabras que se encuentran en el mismo párrafo están relacionadas, se realiza una búsqueda de cada uno de los elementos que se encuentran en el diccionario de candidatos en el Posting List generado en la fase anterior. Una vez que es realizada la búsqueda, cada candidato tiene un conjunto de párrafos en los que se encuentra. A continuación se realiza una intersección de cada uno de los candidatos con el resto, y aquellos que se encuentran relacionados con al menos la mitad de candidatos más uno, se les aumenta un valor en su peso, para así indicar que cumplen con esta segunda característica. Otra característica más es el análisis de similitud semántica mediante la biblioteca gensim y el módulo Word2Vec [12]. Esta herramienta necesita un corpus de tamaño considerable para su funcionamiento. El corpus proporcionado a la herramienta fue el corpus generado mediante la implementación del Crawler en la fase anterior. De manera similar a lo realizado con el Posting List, cada candidato es comparado con el resto de elementos y aquellos candidatos que estén relacionados con los demás se aumentan su valor de peso. Además cada candidato es comparado también con el conjunto de entidades predefinidas por el SemEval. Cuando la medida de similitud encontrada por el modelo generado por Word2Vec entre el candidato y alguna entidad es grande, el valor de peso del candidato es aumentado, esto ya que si es muy similar a una entidad, lo más probable es que sea un aspecto.

La siguiente característica analizada son los patrones encontrados en la primera fase. Estos patrones están conformados por las secuencias de etiquetas de POS (Part of Speech) de los aspectos en los datos de entrenamiento. Para obtener estos patrones cada reseña de los datos de entrenamiento es procesada mediante la herramienta de CLiPS para obtener la etiqueta POS (Part of Speech) de cada

palabra. Posteriormente se forman los patrones, estas son las secuencias de la forma POS_Izquierda + POS_Aspecto + POS_Derecha. Donde POS_Aspecto es la etiqueta o secuencia de etiquetas gramaticales del aspecto; POS_Izquierda y POS_Derecha son las etiquetas gramaticales de la palabra izquierda y derecha al aspecto respectivamente. Para cada reseña, se realiza la búsqueda de los posibles n-gramas que cumplan con el patrón de aspecto detectado. Una vez identificados estos patrones, también son estudiados sus elementos adyacentes izquierdo y derecho. Si cumplen con alguno de los patrones identificados en la fase de pre-procesamiento son agregados al diccionario de candidatos, si este elemento ya se encuentra, su valor de peso es aumentado. Finalmente, la última característica tomada en cuenta es la búsqueda de los candidatos encontrados hasta este momento en la lista de aspectos, del conjunto de datos de entrenamiento. Si es encontrado el candidato en esta lista se aumenta su valor de peso. Ya realizado el análisis de las características mencionadas (Identificación de sustantivos, análisis de contexto, identificación de patrones, identificación de aspectos frecuentes), el criterio de selección de los candidatos es que cumplan con tener un peso mayor o igual a 3, es decir, cumplen con 3 o más características de las mencionadas anteriormente. La metodologìa propuesta se puede ver en la figura:

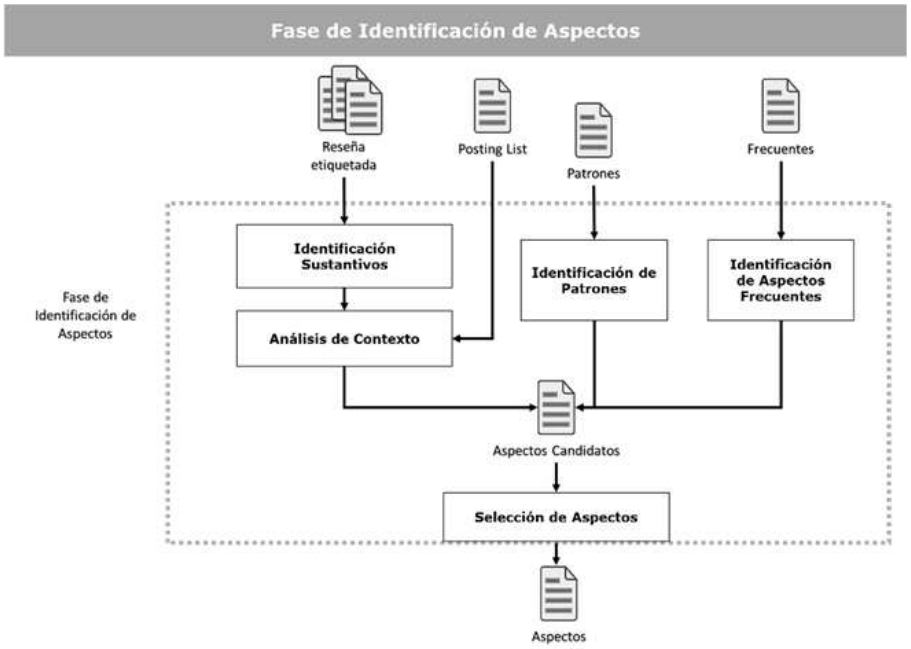

Fig. 2. Metodología basada en tres fases

2.1. Fase de identificación de polaridad

Teniendo ya los candidatos propuestos, las entidades y atributos identificados, la última parte consiste en identificar la polaridad del sentimiento expresado sobre cada aspecto. Para esto, se utilizó un enfoque basado en diccionario, para tener el valor de sentimiento para cada palabra. En esta propuesta se utilizó un diccionario creado en base a los datos de entrenamiento. Se creó mediante el procesamiento de adjetivos, adverbios y verbos en cada enunciado. Los datos de entrenamiento se encuentran etiquetados con su polaridad correcta, por tanto, para cada elemento encontrado en la sentencia se le da un valor de sentimiento en un rango de -1 a 1, donde -1 es muy negativo y 1 es muy positivo. Para el cálculo de este valor se realiza la división entre el número de ocasiones que aparece el elemento para cada polaridad, positiva, negativa y neutra. De estos resultados, el que tenga mayor valor es elegido como la polaridad predominante de esta palabra y el resultado de la división es asignado como valor de sentimiento. Para aquellas palabras que no se encuentren en el conjunto de palabras encontradas en los datos de entrenamiento se utilizó el diccionario SentiWordNet [1] que proporciona un valor numérico de sentimiento para cada palabra.

Posteriormente para detectar la polaridad de cada aspecto encontrado, se realiza el promedio de las polaridades de las palabras de la frase donde el aspecto se encuentra. Se analizan palabras que invierten el valor de polaridad como son “NOT” Cuando este tipo de palabras aparecen, el valor de polaridad de las siguientes palabras de la sentencia es invertido. Palabras como “TOO”, “VERY” entre otras también causan un efecto en las palabras, estas aumentan el valor de polaridad de los siguientes elementos de la sentencia. Al finalizar, si el valor promedio encontrado es positivo y mayor a un rango establecido, la sentencia es clasificada como positiva. De lo contrario si es negativo y menor al rango, es clasificada como negativa. Si el valor promedio es igual a cero o si está dentro del rango establecido es marcada como neutra. El rango mencionado se establece de manera manual, en donde el valor de polaridad identificado es mínimo, lo que implica que el sentimiento expresado sobre un aspecto no es relevante para ser clasificado como positivo o negativo. En adición al promedio de las polaridades, se estudia la polaridad individual de las palabras adyacentes al aspecto identificado. Se analiza el valor de polaridad más alto, además del total de palabras positivas, negativas y neutras. La salida generada por esta fase es el conjunto de aspectos, entidades y atributos con su polaridad identificada. Es to puede observarse en la figura

3. Resultados obtenidos

Una vez desarrollado el modelo, se generaron las salidas necesarias para participar en el Foro de Competencia del SemEval 2016. Para la tarea de la identificación de aspectos, se obtuvo un 50.25

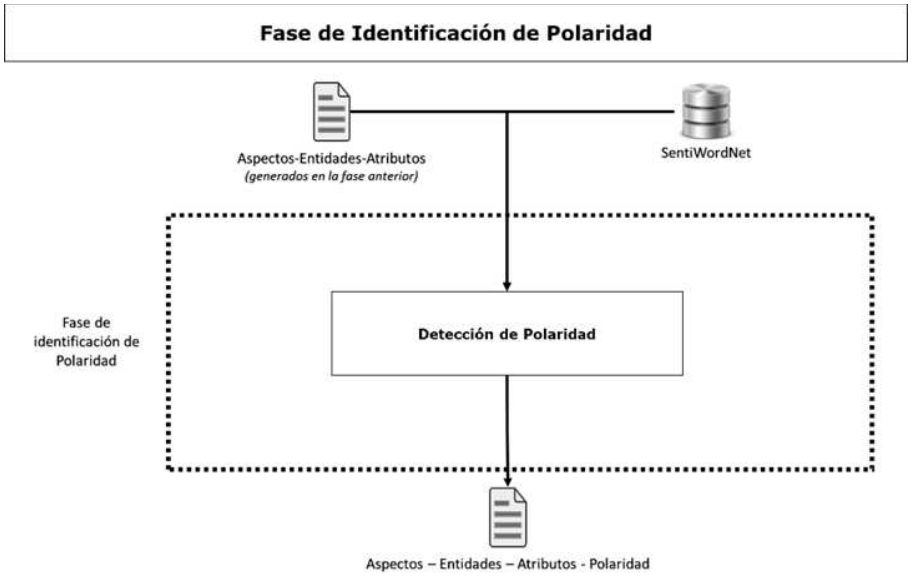

Fig. 3. Metodología basada en tres fases

Tabla 1. Comparación de resultados obtenidos en la identificación de aspectos: Dominio de Restaurantes

Propuestas	**Exactitud**
Mejor propuesta	72.34 %
Nuestra propuesta	50.25 %
Base line	44.07 %

4. Conclusiones obtenidas y trabajo a futuro

Evaluado el modelo propuesto para resolver la tarea del análisis de sentimientos basado en aspectos propuesto por el SemEval, se llegó a las siguientes conclusiones. Se diseñó e implementó un modelo para detectar los aspectos en las reseñas proporcionadas y se detectó la polaridad del sentimiento expresado en la reseña para cada aspecto. Con la implementación del Crawler con los aspectos de entrenamiento en la fase de pre-procesamiento se generaron dos corpus específicos, uno con información relacionada con restaurantes, y otro con datos sobre laptops. Estos corpus son una aportación de gran utilidad para futuras tareas que necesiten trabajar bajo estos contextos o similares. Una vez analizados los resultados obtenidos por el modelo se llegaron a las siguientes conclusiones:

- El modelo se comporta bien en la detección de aspectos, sería de gran utilidad generar un diccionario para mejorar la identificación de aspectos. Este diccionario podría incluir, por ejemplo, nombres de platillos para el caso

Tabla 2. Comparación de resultados obtenidos en la identificación de polaridad: Dominio de Restaurantes

Propuestas	**Exactitud**
Mejor propuesta	88.12 %
Nuestra propuesta	60.88 %
Base line	76.48 %

Tabla 3. Comparación de resultados obtenidos en la identificación de polaridad: Dominio de Laptops

Propuestas	**Exactitud**
Mejor propuesta	82.72 %
Nuestra propuesta	62.79 %
Base line	70.03 %

de restaurantes o nombres de aplicaciones y componentes para el dominio de laptops. Esto debido a que los etiquetadores gramaticales son etiquetadores generales y en ocasiones existen palabras de los contextos de (restaurantes y laptops) que no son etiquetados correctamente.

- Dado que los diccionarios de sentimiento como el utilizado SentiWordNet son diccionarios generales, es necesario generar un diccionario de sentimientos específico para cada contexto, ya que existen palabras como "hot" o "cold" que bajo el contexto general tienen una polaridad y bajo el contexto de comida y restaurantes tienen otro valor totalmente opuesto.
- Analizando los resultados obtenidos al utilizar las funciones brindadas por CLiPS, se concluye que esta herramienta es de gran utilidad, aunque mejora los resultados al apoyarse de otros etiquetadores como los utilizados en el modelo propuesto.
- Se debe mejorar en la tarea de la detección de la polaridad. Analizar problemáticas como el uso del sarcásmo en las críticas que invierten la polaridad del sentimiento expresado y estudiar la manera de mejorar los resultados del modelo.
- La inclusión de la herramienta Word2Vec fue de gran utilidad para mejorar los resultados, ya que proporciona una medida de similitud entre palabras basadas en un contexto dado. Esto mediante el análisis de un corpus especifico. Para mejorar los resultados se podría aumentar el tamaño del corpus de entrada. Finalmente, se pretende continuar con el trabajo en este modelo para futuras participaciones en las tareas del SemEval. Como trabajo futuro se propone lo siguiente:
- La creación de un diccionario de platillos para el dominio de restaurantes y de aplicaciones y componentes para el dominio de laptops.
- La inclusión de los datos de entrenamiento y pruebas del SemEval 2015 y 2016 para mejorar los diccionarios para la identificación de atributos y entidades y la extracción de aspectos.
- Implementar nuevas características que permitan obtener mejores resultados. Estas pueden ser el uso de aprendizaje automático y similitud entre frases.

Agradecimientos. El trabajo realizado con apoyo parcial de CONACYT.

Referencias

1. Baccianella, S., Esuli, A., Sebastiani, F.: Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In: Proc. of LREC (2010)
2. Bird, S., Klein, E., Loper, E.: Natural Language Processing with Python. O'Reilly Media, Inc., 1st edn. (2009)
3. Blinov, P., Kotelnikov, E.: Blinov: Distributed representations of words for aspect-based sentiment analysis at semeval 2014. In: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). pp. 140–144. Association for Computational Linguistics and Dublin City University, Dublin, Ireland (August 2014)
4. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 168–177. KDD '04, ACM, New York, NY, USA (2004)
5. Kiritchenko, S., Zhu, X., Cherry, C., Mohammad, S.: Nrc-canada-2014: Detecting aspects and sentiment in customer reviews. In: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). pp. 437–442. Association for Computational Linguistics and Dublin City University, Dublin, Ireland (August 2014), http://www.aclweb.org/anthology/S14-2076
6. Manning, C.D., Schütze, H.: Foundations of Statistical Natural Language Processing. MIT Press, Cambridge, MA, USA (1999)
7. de Marneffe, M.C., MacCartney, B., Manning, C.D.: Generating typed dependency parses from phrase structure parses. In: Proc.of LREC. pp. 449–454 (2006)
8. Miller, G.A., Beckwith, R., Fellbaum, C., Gross, D., Miller, K.: Wordnet: An on-line lexical database. International Journal of Lexicography 3, 235–244 (1990)
9. Nair, V.G.: Getting Started with Beautiful Soup. Packt Publishing (2014)
10. Nasukawa, T., Yi, J.: Sentiment analysis: Capturing favorability using natural language processing. In: Proceedings of the 2Nd International Conference on Knowledge Capture. pp. 70–77. K-CAP '03, ACM, New York, NY, USA (2003)
11. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up?: Sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10. pp. 79–86. EMNLP '02, Association for Computational Linguistics, Stroudsburg, PA, USA (2002)
12. Řehůřek, R., Sojka, P.: Software Framework for Topic Modelling with Large Corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks. pp. 45–50. ELRA, Valletta, Malta (May 2010)
13. Schouten, K., Frasincar, F., de Jong, F.: Commit-p1wp3: A co-occurrence based approach to aspect-level sentiment analysis. In: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). pp. 203–207. Association for Computational Linguistics and Dublin City University, Dublin, Ireland (August 2014)

Comparison of Automatic Keyphrase Extraction Systems in Scientific Papers

Jesús Ernesto Padilla Camacho, Yulia Ledeneva, René Arnulfo García Hernández

Autonomous University of the State of Mexico,
State of Mexico, Mexico

jernestopc1@gmail.com, yledeneva@yahoo.com, renearnulfo@hotmail.com

Abstract. Nowadays the amount of digital information that found in internet has considerably increased that is why online search is needed to automatically find the corresponding documents. These documents must be verified in order to know whether they contain the required information. A way to simplify the online search is using keywords or keyphrases since they act as filters within a search field. The paper presents the comparison of automatic keyphrases extraction systems based on a collection of scientific papers used in task 5 of SemEval-2010 which calls "Automatic keyphrase extraction from scientific articles". In the experimental section, the results are presented for installable and online systems. We found systems that can match better the author-, reader-, and combined-assigned keyphrases with the keyphrases proposed by an expert. Finally, the obtained results are compared to the results obtained in task 5 of SemEval-2010.

Keywords: Automatic keyphrases extraction, task 5 SemEval-2010, KEA, Alchemy, Wordstat, Extractor.

1 Introduction

At present time the usage of data is a factor of great importance in the public and private sectors. With the constant increase of digital information it needs to be organized for the usage. Nowadays with the technology advances, the searching of information has been facilitated. The keyphrases helps in the information retrieval task because they are very useful for searching information in a big data collection and act as filter to show the most important topics that are described by the author. The keyphrases are the union of words that represent the main ideas of text and provide a brief perception of its content [1-5].

The automatic selection of keyphrases that best describe a text is called Automatic Keyphrase Extraction (AKE). That is to say, the AKE is responsible to perform all the process that is made by a professional indexer, since executing the process automatically reduces factors as the cost of hiring an expert on the subject and the time involved. Witten [3] mentions that the keyphrases usually are elected manually in many academic contexts. The authors assign keywords at the documents they write. The professional indexers usually elect phrases from a "controlled vocabulary" that is relevant for the domain. However, the great majority of documents come without

keyphrases, and manually assigning it is a tedious process that requires a specialized knowledge.

The organization of paper is as follows: in section 2, the related works to AKE are mentioned. In the section 3, the proposed frame work for evaluating AKE systems is described. In the section 4, the dataset, the evaluation tool and the results of experimentation are presented. In the section 5, the conclusions of paper are presented.

2 Related Work

In 2010, Kim et al. [1, 2] perform the shared task "Task 5: Automatic Keyphrase Extraction from Scientific Articles" that was included in the SemEval-2010. The purpose was to develop AKE systems from scientific papers and compare the list of keyphrases proposed by each competitor system with the keyphrases that were assigned by humans to each of the scientific papers. The system that won the best result in the task was the system HUMB [7] with F-score of 27.5% in the combined-assigned configuration.

HUMB is supervised approach that analyzes the structure of document (abstract, conclusions, and references). The selection of candidates that implements is the extraction of n-grams up to 5 words, elimination of candidates that started or terminated with stopwords, filtering of mathematical symbols. The classification of candidates is done by a decision tree. Also the terminology databases GRISP [8] and Wikipedia [9] are used.

Nguyen [10] participates in the task 5 SemEval-2010 with the system WINGNUS. WINGNUS is a supervised approach, one of the main characteristics that employ for the keyphrases extraction is the logic structure of document, to make less text to analyze. The sections are identified where is the most probable is to find the keyphrases. They consider that these sections are abstract, introduction and conclusions. For the classification of candidates employs 19 syntactic functions of which the best result is obtained with the functions such as: *tf* x *idf*, term frequency, substrings frequency, first occurrence and length of the phrase.

El-Beltagy [11] participates in the task 5 SemEva-2010 with the system KP-miner. KP-miner is an unsupervised approach that extracts keyphrases from text in Arab and English. The process consists of three steps: 1.- Selection of candidates where the words are filtered which are not separated by punctuation signs or stopwords, also the frequency of phrase and first occurrence is included. 2.- Weight calculation: term weight, term frequency, IDF weighting, increase factor and term position. 3.- Selected list of final candidate for keyphrases: this is an optional characteristic of the system to refine the candidates.

Bernend [12] participates in the task 5 SemEvla-2010 with the system SZTERGAK. SZTERGAK is a supervised approach. The selection of candidates that employs is the extraction n-grams up to 4 words, the characteristics are grouped in four categories: 1.- Sentence level (length word and POS pattern). 2.- Document level (such characteristics as: acronomity, PMI Sintactic). 3.- Corpora level (*tf-idf* and *keyphraseness*). 4.- External knowledge: use of Wikipedia.

Pianta [13] participates with the system KX. This it is an unsupervised approach. KX employs four steps for the selection of candidates to extract n-grams up to 4 words:

three at corpora level and one extracts the specific document information. For the classification of candidates employs the next characteristics: *idf*, length phrase, position of first occurrence, subsumption and boosting.

The state-of-the-art of AKE systems that not included in task 5 are presented below.

Witten et al. [3] create an algorithm which calls KEA. The proposed algorithm is a supervised approach uses the technique of *Naive Bayes*, which from training data creates a training model that can extract the keyphrases of new documents. KEA employs 2 characteristics: *tf-idf* and first occurrence of phrase.

Turney [5] presents the results of comparison between an extraction model based in a genetic algorithm and an implementation of C4.5 decision trees. Turney informs that genetic algorithm issues better keywords than decision trees.

Mihalcea [6] presents a classification model based on an unsupervised graph that uses the co-occurrence and relation between words that added to the graph to give weights to the vertices. TextRank perform two tasks inside of the information retrieval that are: keyphrase extraction and keyphrase extraction for text summarization.

Medelyan [14] presents Maui, this is a variant of KEA. This is a supervised algorithm for the automatic indexing, uses semantic information extracted from Wikipedia which uses external resources to obtain the best keyphrase extraction based in the titles from Wikipedia.

3 Framework

The dataset of task 5 of SemEval2010 is used. First, the pre-processing is applied. Second, the AKE based on the standard configuration of parameters is performed. Third, Porter stemmer algorithm [15] is applied to obtain the evaluation format. Forth, the evaluation is performed with the tool that evaluates the results, same that it is used in the task 5 of SemEval-2010 (performance.pl). Finally, the systems are compared to the results which are presented in task 5. The evaluated systems are divided in two categories: installable and online systems.

Online systems are those that are run from a web page. The online systems we use for evaluation are mentioned as follows: Alchemy [16] is a commercial system it belongs to the products of IBM family. It offers the AKE as well as entities, text sentiment, classifies the relevance of results, returns the results in different format and it works with a great range of languages. Skyttle [17] is a commercial system for the AKE and text sentiment, it works only in English. Fivefilters [18] is a terms extractor of open source that returns the most important terms. The parameters are: maximum of results, special formats of results, maximum of words per term. It works only in English. Genia Tagger [19] is a commercial terms extractor designed for texts of biomedical area. We use it in this work for learning the performance in other domain. Tree Tagger [19] is a commercial terms extractor that returns the main terms of an analyzed text. It works only in English. Transalated Labs [20] is a terms extractor for identification of the main terms in one text. It works in French, Italian and English.

Installable systems are those that run locally in computers. The installable systems we use for evaluation as follows: KEA [21] is an open source system of supervised approach that from training dataset can perform automatic keyphrase extraction. The parameters are length phrase, minimum occurrence, vocabulary name. It works in

English, Spanish and French. Extractor [22] is commercial system which extracts keyphrases from a text. The parameters are the number of keyphrases for extraction and list of stopwords. It works in English, Spanish, French, German, Japanese and Korean. Wordstat [23] is a commercial system that counts with tools for the text processing. For the automatic keyphrase extraction, the parameters are length phrase, minimum of occurrence, it works with a great range of languages. TexLexAn [24] is an open source system that incorporates different applications as automatic text summarization, plagiarism detection, keyword extraction. It works in English, Spanish, French, German and Italian.

4 Experimental Results

4.1 Dataset

The dataset used is a collection of scientific articles from task 5 of SemEval-2010. The articles come from the digital library ACM. The distribution of the 4 areas that contains the corpora SemEval-2010 [1]. There are three assignments of golden keyphrases: 1.-Author: keyphrases that have been assigned by the authors of scientific articles by defect. 2.-Reader: keyphrases that were assigned by the readers of scientific articles. 3.-Combined: combination between keyphrases of author and reader.

The systems are presented by the top 5, 10, 15 keyphrases and ranked according to F-score by the top 15 keyphrases as originally in SemEval-2010.

In this paper, the evaluation is implemented using a standard configuration with the objective of measuring the performance of the systems under the following parameters:

Number of keyphrases to extract: a list of 15 keyphrases are extracted for each of 100 scientific articles from SemEval2010. **Minimum length**: keyphrase can be considered of the length of one word. **Maximum length**: based on the system HUMB [7], the maximum number of words that can contain a keyphrase is 5. This is done with the purpose of including the major amount of keyphrases with 4 and 5 words, which mostly occurred in the corpora. There are also longer keyphrases, however contains stopwords. **Frequency**: the systems that have this parameter are left by default for the systems that requires it.

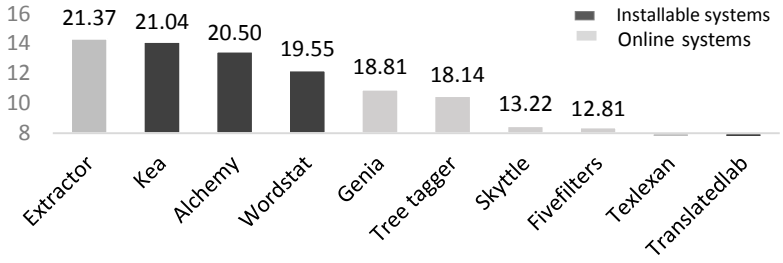

Fig. 1. The performance of author-assigned keyphrases systems (F-score, top 15)

4.2 Results of the Author-assigned Keyphrases

In the author-assigned keyphrases, *KEA* is located in the top 5 as the system with higher result of Precision 15.20%, Recall 19.64% and F-score of 17.14%. For the top 10, KEA again have the highest values of Precision 11.20%, Recall 28.94% and F-score 16.15%. In the top 15, Extractor is positioned in first place for the author-assigned with Precision 9.0%, Recall 34.88% and F-score 14.31%. In table 2, the systems are ranked by the F-score obtained in top 15 keyphrases where the highest values are marked in the top 5, 10 and 15 (see table 1 and figure 1).

Table 1. Results of the systems in the author-assigned (top 15)

		top 5			top10			top15		
System	Rank	P	R	F	P	R	F	P	R	F
Extractor	1	14.80	19.12	16.68	10.40	26.87	15.00	**9.00**	**34.88**	**14.31**
Kea	2	**15.20**	**19.64**	**17.14**	**11.20**	**28.94**	**16.15**	8.87	34.37	14.10
Alchemy	3	14.60	18.86	16.46	10.20	26.36	14.71	8.47	32.82	13.47
Wordstat	4	14.40	18.60	16.23	10.00	25.84	14.42	7.67	29.72	12.19
Genia	5	14.00	18.09	15.78	9.50	24.55	13.70	6.87	26.61	10.92
Tree tagger	6	13.40	17.31	15.11	9.10	23.51	13.12	6.60	25.58	10.49
Skyttle	7	8.20	10.59	9.24	6.40	16.54	9.23	5.33	20.67	8.47
Fivefilters	8	6.00	7.75	6.76	5.50	14.21	7.93	5.27	20.41	8.38
Texlexan	9	5.80	7.49	6.54	4.80	12.40	6.92	4.00	15.50	6.36
Translatedlab	10	5.60	7.24	6.32	4.00	10.34	5.77	3.20	12.40	5.09

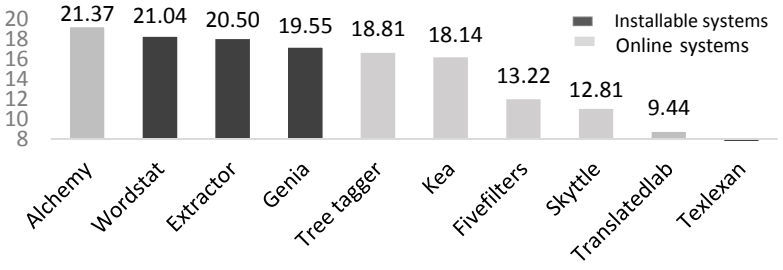

Fig. 2. The performance of reader-assigned keyphrases systems (F-score, top 15)

4.3 Results of Reader-assigned Keyphrases

In the reader-assigned keyphrases, Wordstat is located in the top 5 as system with the highest percentage in Precision 26.20%, Recall 10.88% and F-score 15.38%. For the top 10, Wordstat and Alchemy have the same values in Precision 20.10%, Recall 16.69% and F-score 8.24%. In the top 15, Alchemy with Precision 17.40%, Recall

21.68% and F-score of 19.31%, is positioned in first place of the reader-assigned keyphrases. In table 3, the results are shown where the systems are ranked by the F-score obtained in the top 15 and more higher values are marked in the top 5, 10 and 15 (see table 2 and figure 2).

Table 2. Results of the systems of the reader-assigned keyphrases (top 15)

		Top5			top10			top15		
Systems	Rank	P	R	F	P	R	F	P	R	F
Alchemy	1	25.20	10.47	15.38	**20.10**	**16.69**	**18.24**	**17.40**	**21.68**	**19.31**
Wordstat	2	**26.20**	**10.88**	**15.38**	**20.10**	**16.69**	**18.24**	16.53	20.60	18.34
Extractor	3	19.00	7.89	11.15	17.20	14.29	15.61	16.33	20.35	18.12
Genia	4	24.40	10.13	14.32	18.90	15.70	17.15	15.53	19.35	17.23
Treetrager	5	25.20	10.47	14.79	18.50	15.37	16.79	15.07	18.77	16.72
Kea	6	20.00	8.31	11.74	17.00	14.12	15.43	14.67	18.27	16.27
Fivefilters	7	13.20	5.48	7.74	11.70	9.72	10.62	10.87	13.54	12.06
Skyttle	8	12.40	5.15	7.28	10.80	8.97	9.80	10.00	12.46	11.10
Translatedlab	9	11.20	4.65	6.57	9.10	7.56	8.26	7.93	9.88	8.80
Texlexan	10	10.00	4.15	5.87	7.60	6.31	6.90	6.40	7.97	7.10

4.4 Results of Author- and Reader-, Combined-assigned Keyphrases

In the combined keyphrases, Wordstat is located in the top 5 as the system with the highest result in Precision 32.20%, Recall 10.98% and F-score of 16.38%. The same result, for the top 10, Wordstat is positioned in first place with 24.5% of Precision, Recall 16.71% and F-score of 19.87%. Alchemy in the top 15 is positioned in first place in the combined keyphrases with Precision 21.13%, Recall of 21.62% and F-score of 21.37%. In table 4, the results are shown where the systems are ranked by the F-score obtained in the top 15 keyphrases and more higher values are marked in the top 5, 10 and 15 (see table 3 and figure 3).

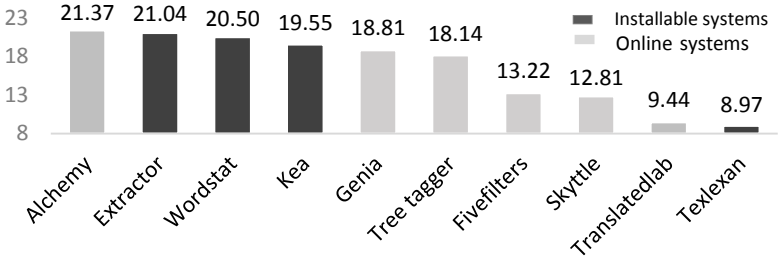

Fig. 3. The performance of combined keyphrases systems (F-score, top 15)

Table 3. Results of the systems in the combined keyphrases (top 15)

		top 5			top10			top15		
Systems	Rank	P	R	F	P	R	F	P	R	F
Alchemy	1	31.20	10.64	15.87	24.40	16.64	19.79	**21.13**	**21.62**	**21.37**
Extractor	2	27.00	9.21	13.73	22.10	15.08	17.93	20.80	21.28	21.04
Wordstat	3	**32.20**	**10.98**	**16.38**	**24.50**	**16.71**	**19.87**	20.27	20.74	20.50
Kea	4	27.80	9.48	14.14	22.80	15.55	18.49	19.33	19.78	19.55
Genia	5	29.60	10.10	15.10	23.00	15.69	18.65	18.60	19.03	18.81
Treetrager	6	30.00	10.20	15.30	22.30	15.21	18.08	17.93	18.35	18.14
Fivefilters	7	16.40	5.59	8.34	14.40	9.82	11.68	13.07	13.37	13.22
Skyttle	8	16.20	5.53	8.25	13.90	9.48	11.27	12.67	12.96	12.81
Translatedlab	9	14.00	4.77	7.12	10.80	7.37	8.76	9.33	9.55	9.44
Texlexan	10	13.40	4.57	6.82	10.70	7.30	8.68	8.87	9.07	8.97

4.5 Comparison of Results with Systems in SemEval-2010

The comparison of the results in this evaluation of the task 5 of SemEval-2010 are presented, with the objective of learning if the systems actually present better performance that already evaluated before.

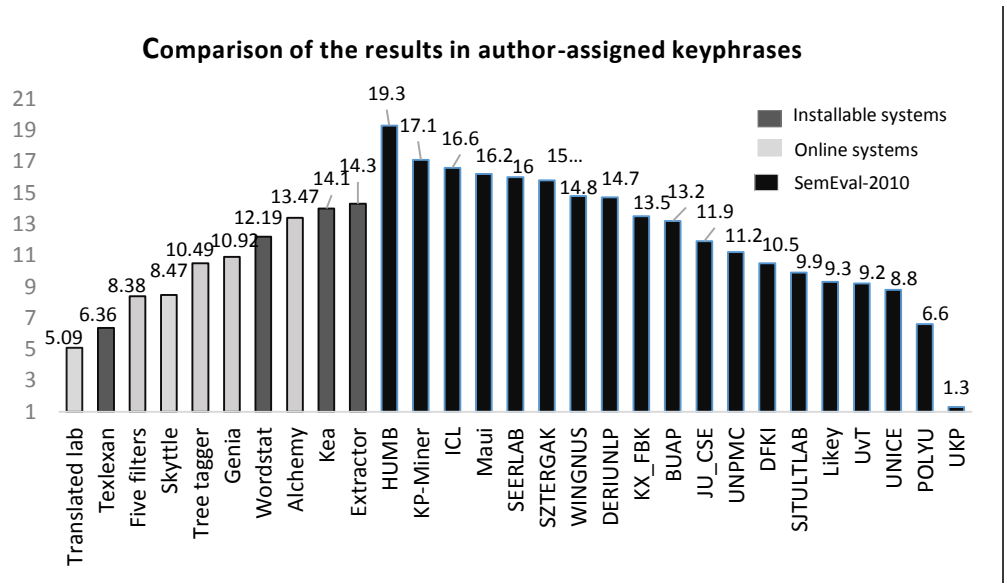

Fig. 4. The performance of the evaluated systems and the systems participated in SemEval-2010 of the author-assigned keyphrases, ranked by F-score of the top 15

Comparison of the results in the author-assigned keyphrases

In this paper, the obtained results of the system *Extractor* of author-assigned keyphrases is 14.31%. DERIUNLP with 14.70% has the similar result in SemEval-2010, while lowest result in this evaluation is *Translatedlab* with 5.09%. In the top 15 in SemEval-2010, UKP obtained 1.3%. In figure 4, the best reached result in this evaluation is *Extractor* 14.31% and in SemEval-2010 is HUMB 19.3%. The bars of color strong gray belong to the installable systems and the bars of color light gray to the online systems that are compared in this paper, while the black bars belong to the systems participated in the task 5 of SemEval-2010 (see figure 4).

Comparison of the results in the reader-assigned keyphrases

In the reader-assigned keyphrases, the system with the highest result is *Alchemy* with 19.31%, its result is similar to DERIUNLP with 19.5% and DFKI with 19.3% in SemEval-2010. The system with the lowest result in this evaluation is *TexLexAn* with 7.1%, while in SemEval-2010 is UKP with 5.2%. In figure 5, the best obtained result in this evaluation is *Alchemy* with 19.31% and in SemEval-2010 is HUMB with 23.5%. The bars of color strong gray belong to the installable systems and the bars of color light gray to the online systems that are compared in this paper, while the black bars belong to the systems participated in the task 5 of SemEval-2010 (see figure 5).

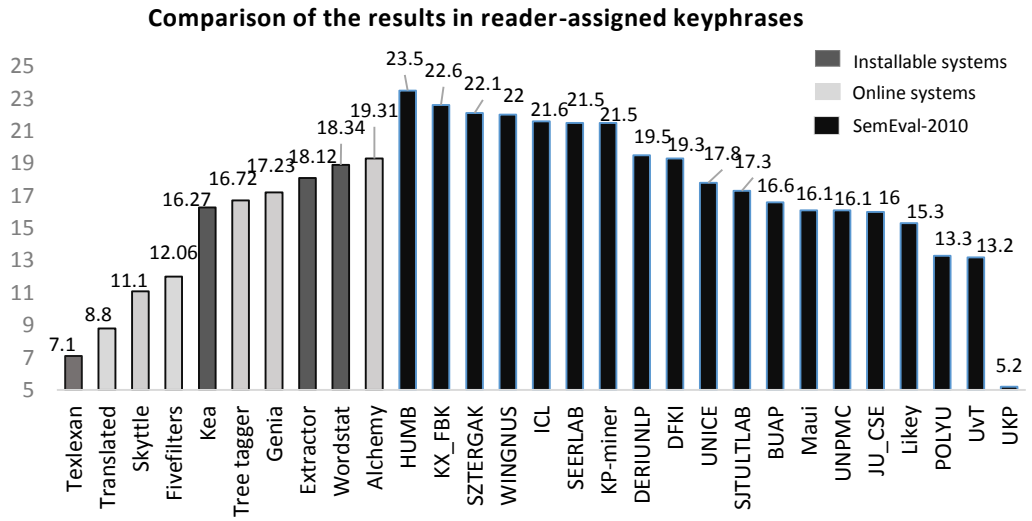

Fig. 5. The performance of the evaluated systems and the systems participated in SemEval-2010 of the reader-assigned keyphrases, ranked by F-score of the top 15

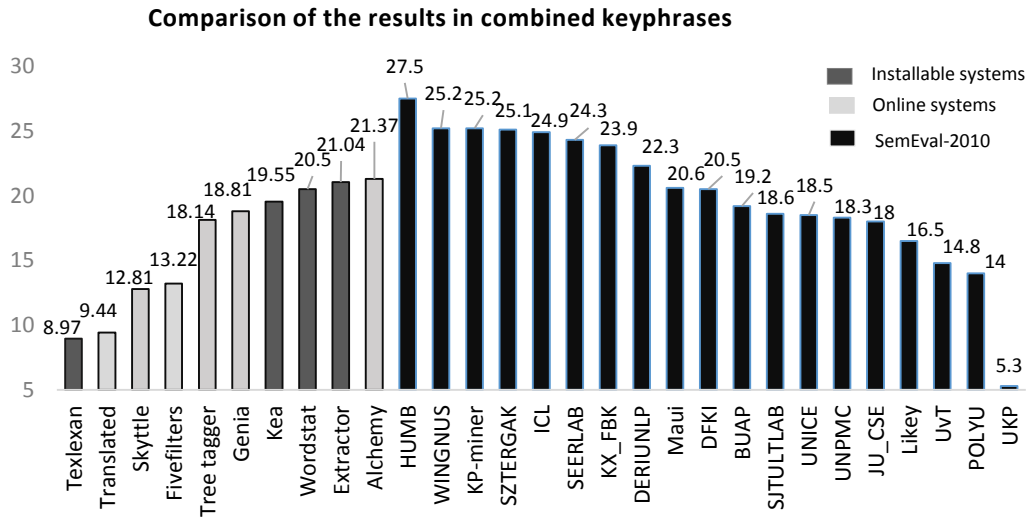

Fig. 6. The performance of the evaluated systems and the systems participated in SemEval-2010 of the combined-assigned keyphrases, ranked by F-score of the top 15

Comparison of the results in the combined keyphrases

In the combined-assigned keyphrases, the system with the highest result in this evaluation is Alchemy with 21.37%. The result can be compared to DERIUNLP with 22.30% and Maui with 20.60%. The system with the lowest result in this evaluation is TexLexAn with 8.97% while in SemEval-2010 is UKP with 5.3%. In figure 6, the best obtained result in this evaluation is Alchemy with 21.37% and in SemEval-2010 is HUMB with 27.5%. The bars of color strong gray belong to the installable systems and the bars of color light gray to the online systems that are compared in this paper, while the black bars belong to the systems participated in the task 5 of SemEval-2010 (see figure 6).

The comparison of the obtained results showed the ranking of the state-of-the-art AKE systems with the already evaluated systems of SemEval-2010.

The results of the systems presented in SemEval-2010 are superior for some systems and are equal for another systems evaluated in this work. We expected that systems evaluated in this paper would present better performance that previously evaluated, considered the time that has passed since 2010.

Also, the presented results show that some terms extraction systems obtain better results than the keyphrase extraction systems.

5 Conclusions

In this paper, the evaluation of systems that automatically extract keyphrases is presented over the commercial free systems that are available in internet for the usage and download. The contribution of the paper is to present the performance of the-state-

of the-art systems and compare the performance with the systems evaluated in SemEval-2010. According to the ranking of systems by the three assignments that contains the gold keyphrases, the system Extractor obtained the first place in the author-assigned while the system Alchemy obtained the first place in the reader- and combined-assigned.

The future work is to test the state-of-the-art systems over other dataset with author- and reader-assigned keyphrases. Other idea is to learn the performance of the systems in different domains. Also, syntactic n-grams [25, 26] and maximal frequent sequences [27-29] will be tested.

Acknowledgment. The work was done with partial support of the Mexico government (CONACyT, SNI, and UAEMex). The authors acknowledge to the Autonomous University of the State of Mexico (UAEMex) for the support.

References

1. Kim, S. N., Medelyan, O., Kan, M. Y., Baldwin, T.: Semeval-2010 task 5: Automatic keyphrase extraction from scientific articles. In Proceedings of the 5th International Workshop on Semantic Evaluation, Association for Computational Linguistics (2010)
2. Kim, S.N., Medelyan, O., Kan, M.Y., Baldwin, T.: SemEval-2010 Task 5: Automatic Keyphrases Extraction from Scientific Articles. Language resources and evaluation, Vol. 47, Issue 3 (2013)
3. Witten, I.H., Paynter, G.W., Frank, E., Gutwin, C., Nevill-Manning, C.G.: KEA: Practical automatic keyphrase extraction. Proceedings of the fourth ACM conference on Digital libraries, pp. 254–255, ACM (1999)
4. Medelyan, O., Witten, I.H.: Thesaurus based automatic keyphrase indexing. Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries, pp. 296–297, ACM (2006)
5. Turney, P.D.: Learning algorithms for keyphrase extraction. Information Retrieval, Vol. 2, No. 4, pp. 303–336 (2000)
6. Mihalcea, R., Tarau, P.: TextRank: Bringing order into texts. Association for Computational Linguistics (2004)
7. Lopez, P., Romary, L.: HUMB: Automatic key term extraction from scientific articles in GROBID. Proceedings of the 5th international workshop on semantic evaluation Association for Computational Linguistics. pp. 248–251 (2010)
8. Lopez, P., Romary, L.: GRISP: A Massive Multilingual Terminological Database for Scientific and Technical Domains. Seventh international conference on Language Resources and Evaluation (LREC), Valletta, Malta (2010)
9. Wikipedia Database URL: https://en.wikipedia.org/wiki/Wikipedia:Database_download. Consultado 05/3/16.
10. Nguyen, T.D., Luong, M.T.: WINGNUS: Keyphrase extraction utilizing document logical structure. Proceedings of the 5th international workshop on semantic evaluation, pp. 166–169, Association for Computational Linguistics (2010)
11. El-Beltagy, S.R., Rafea, A.: Kp-miner: Participation in Semeval-2. Proceedings of the 5th international workshop on semantic evaluation, pp. 190–193, Association for Computational Linguistics (2010)
12. Berend, G., Farkas, R.: SZTERGAK: Feature engineering for keyphrase extraction. Proceedings of the 5th international workshop on semantic evaluation, pp. 186–189, Association for Computational Linguistics (2010)

13. Pianta, E., Tonelli, S.: KX: A flexible system for keyphrase extraction. Proceedings of the 5th international workshop on semantic evaluation, pp. 170–173, Association for Computational Linguistics (2010)
14. Medelyan, O., Frank, E., Witten, I.H.: Human-competitive tagging using automatic keyphrase extraction. Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Vol. 3, pp. 1318–1327, Association for Computational Linguistics (2009)
15. Porter Stemming algorithm, http://tartarus.org/martin/, Consulted 05/3/16
16. AlchemyAPI Keyword Extraction API. Consulted 23/12/15. http://www.alchemyapi.com/products/demo/alchemylanguage.
17. Skyttle. URL: http://www.skyttle.com/demoin. Consulted 23/12/15.
18. Fivefilters Term Extraction. http://fivefilters.org/term-extraction/ Consulted 23/12/15.
19. Termine. Termine web demonstration. http://www.nactem.ac.uk/software/termine, Consultado 23/12/15.
20. Translated Labs. Consulted 23/12/15. http://labs.translated.net/terminology-extraction/.
21. KEA. Keyphrase extraction algorithm. Consulted 23/12/15. http://www.nzdl.org/Kea/.
22. Extractor. Extractor Live Content Demonstration. Consulted 23/12/15. http://www.dbitech.com/trials/dbi_TrialDownloads.aspx.
23. Wordstat 7. Software de análisis de contenido y minería de texto. Consulted 23/12/15. http://provalisresearch.com/es/products/software-de-analisis-de-contenido/.
24. TexLexAn. TexLexAn Analyze, Classify and Summarize any text. Consulted 23/12/15. http://texlexan.sourceforge.net/
25. Sidorov, G.: Syntactic dependency based n-grams in rule based automatic English as second language grammar correction. International Journal of Computational Linguistics and Applications, Vol. 4, No. 2, pp. 169–188 (2013)
26. Sidorov, G.: N-gramas sintácticos no-continuos. Polibits, Vol. 48, pp. 69–78 (2013)
27. García-Hernández, R.A., Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A.: A new algorithm for fast discovery of maximal sequential patterns in a document collection. Computational Linguistics and Intelligent Text Processing, pp. 514–523, Springer Berlin Heidelberg (2006)
28. Ledeneva, Y., Gelbukh, A., García-Hernández, R.A.: Terms derived from frequent sequences for extractive text summarization. Computational Linguistics and Intelligent Text Processing, Springer Berlin Heidelberg, pp. 593–604 (2008)
29. Ledeneva, Y., García-Hernández, R.A., Gelbukh, A.: Graph ranking on maximal frequent sequences for single extractive text summarization. Computational Linguistics and Intelligent Text Processing, pp. 466–480, Springer Berlin Heidelberg (2014)

Impreso en los Talleres Gráficos
de la Dirección de Publicaciones
del Instituto Politécnico Nacional
Tresguerras 27, Centro Histórico, México, D.F.
septiembre de 2016
Printing 500 / Edición 500 ejemplares