Técnicas basadas en grafos para la categorización de tweets por tema *

Graph-based Techniques for Topic Classification of Tweets

Héctor Cordobés[1] **Antonio Fernández Anta**[1] **Luis Felipe Núñez**[1]
Fernando Pérez[2] **Teófilo Redondo**[3] **Agustín Santos**[1]

[1] Institute IMDEA Networks, Madrid, Spain
[2] U-tad, Madrid, Spain
[3] Factory Holding Company 25, Madrid, Spain

Resumen: La clasificación de textos por tema es uno de los problemas más interesantes del procesamiento del lenguaje natural (NLP). En este trabajo proponemos técnicas basadas en similitud de grafos para la determinación de un tema dado un mensaje Twitter.
Palabras clave: Categorización por tema, grafos, procesamiento de lenguaje natural

Abstract: Topic classification of texts is one of the most interesting challenges in Natural Language Processing (NLP). In this work we present a technique based on graph similarity to classify Twitter messages as being related to a specific topic.
Keywords: Topic classification, text classification, graphs, natural language processing

1. Introducción

Es este documento presentamos las técnicas usadas para obtener los resultados experimentales que hemos enviado a la edición de 2013 del Taller de Análisis de Sentimientos en la SEPLN (*Workshop on Sentiment Analysis at SEPLN*), TASS 2013. Como en años anteriores, los organizadores de este taller han preparado y hecho público un conjunto de datos de evaluación que han permitido a los participantes en el taller evaluar los sistemas de procesamiento de lenguaje natural que han desarrollado. En particular, de las cuatro tareas planteadas, nosotros hemos participado en la *Task 2: Topic classification*. Para esta tarea, los organizadores han suministrado un conjunto de mensajes en castellano procedentes de Twitter (tweets). Algunos de estos mensajes han sido previamente clasificados en un conjunto prefijado de categorías (política, economía, etc.), y el resto deben ser clasificados por los sistemas desarrollados por los participantes.

La clasificación automática de textos es una de las aplicaciones clave en el procesamiento del lenguaje natural. En particular, en el contexto de nuestro proyecto (denominado SOCAM) la clasificación automática de textos es uno de los elementos más representativos. El objetivo último de dicho proyecto es el desarrollo de un ecosistema para dispositivos conectados a internet, fundamentalmente ideado para móviles y tabletas digitales. El Taller de Análisis de Sentimientos de la SEPLN TASS 2013 ha supuesto una oportunidad para aplicar dicho conocimiento en los campos del análisis de opinión y la clasificación por tema, tan relevantes a la hora de comprender mejor al usuario y sus necesidades (por ejemplo, analizando los temas más frecuentes en los comentarios y valoraciones de los usuarios).

Nuestro trabajo aborda el problema de categorización por tema desde un enfoque orientado a grafos, que hasta lo que hemos podido saber es totalmente novedoso.

2. Descripción de los principios básicos del categorizador

El principio básico en que se basan todas nuestras técnicas es que todo texto (*tweet* en

* Financiado en parte por el proyecto de investigación SOCAM financiado por el Ministerio de Industria, Energía y Turismo, a través de su programa AVANZA, liderado por las empresas Zed Worldwide S.A. y Factory Holding Company 25 S.L. (FHC25).

este caso) se puede representar mediante un grafo. Para un texto dado, nuestra propuesta toma como vértices del grafo las palabras del texto (considerando en realidad la raíz de las mismas) y crea aristas con peso entre las palabras. Hemos considerado distintas formas de asignar peso a las aristas. Una opción simple es que el peso represente la frecuencia con la que ambas palabras aparecen juntas dentro del texto. Otra opción más sofisticada (y compleja) es que esta frecuencia venga ponderada por la distancia (en un árbol sintáctico) entre las dos palabras. Existen otras alternativas para la construcción del grafo que consideramos interesantes para ser analizados en futuros trabajos (en particular las alternativas basadas en grafos dirigidos).

Sabiendo cómo construir un grafo para cada texto (*tweet*), la primera hipótesis en que se basa nuestro sistema es que los grafos que pertenecen a textos del mismo tema forman estructuras representativas (grafo del tema). Para la clasificación de un texto, se busca similitud entre el grafo formado por el texto a clasificar y los diferentes grafos representativos. Por ello nuestro trabajo aplica una técnica de similitud de grafos para detectar el tema de un texto.

Para nuestro estudio hemos decidido construir el grafo de referencia de un tema como la unión de todos los grafos generados a partir de los textos del mismo tema. Esta decisión se basa en la segunda hipótesis de nuestro trabajo, que es que las palabras se relacionan con intensidades muy diferentes según el tema que tratan. Por ejemplo, la palabras *Presidencia* y *Congreso* tendrán una fuerte relación cuando la temática sea *Política*. Esas palabras pueden no aparecer o tener una relación muy débil en otros temas (p.e. *Fútbol*). Por lo tanto, es posible crear un grafo unión diferente según el tema. El proceso de construccion de los grafos de referencia se representa en la figura 1.

Entonces, utilizando los textos de entrenamiento (los *tweets* pre-clasificados), nuestro sistema construye un conjunto de grafos de referencia. La unión de grafos simplemente une los conjuntos de vértices y suma los pesos de las aristas comunes. Cuando se desea clasificar un nuevo texto, se busca el grafo de referencia que más similitud tiene con el grafo del texto a clasificar. La gráfica 2 representa este proceso.

El mecanismo básico descrito anteriormente abre un amplio espectro de posibilidades y de métodos que pueden ser combinados de múltiples formas. El primer paso del mecanismo es la construcción del grafo a partir de un texto. Como hemos dicho, en nuestro trabajo hemos explorado varias alternativas, que usan diferentes criterios de asignación de pesos a los enlaces. De la misma forma, hemos utilizado distintos criterios para medir la similitud de un grafo a clasificar con un grafo de referencia. Detallamos los métodos utilizados en las siguientes secciones.

3. Implementación del categorizador

En esta sección describimos cómo se ha implementado el categorizador desarrollado, y en particular cómo se han implementado las técnicas descritas en la sección anterior. En la Sección 3.1 se describe el preprocesado al que se someten los textos antes de usarlos para construir los grafos asociados. La Sección 3.2 describe cómo se construyen los grafos de referencia. Finalmente, en la Sección 3.3 se describe cómo se identifica el tema de un nuevo mensaje.

3.1. Preprocesado del texto

Como paso previo a la construcción y análisis de los grafos es necesario realizar un procesado previo de tratamiento de los textos. Este paso es habitual en las técnicas de procesamiento de lenguaje natural. En éste el texto se corrige, analiza y descompone en elementos más simples. En nuestro trabajo hemos utilizado los diccionarios de Hunspell para la corrección ortográfica del texto. También se ha utilizado un diccionario de abreviaturas y símbolos de SMS (diccionario SMS) que ya formó parte del sistema que desarrollamos para TASS 2012 (Fernández Anta et al., 2013). Adicionalmente, se ha utilizado Freeling (Padró et al., 2010) para la lematización (extraer el lema) de las palabras, teniendo en cuenta la desambiguación automática de lemas según la función sintáctica. Freeling se utiliza también para obtener el árbol sintáctico de los mensajes, y poder así calcular en ellos las distancias entre palabras, que serán utilizadas en secciones siguientes.

Un paso relevante de nuestro preproceso ha sido el reconocimiento de nombres de entidades (*Named Entity Recognition*, NER). El objetivo de este paso ha sido disponer de mecanismos que determinen y unifiquen en un

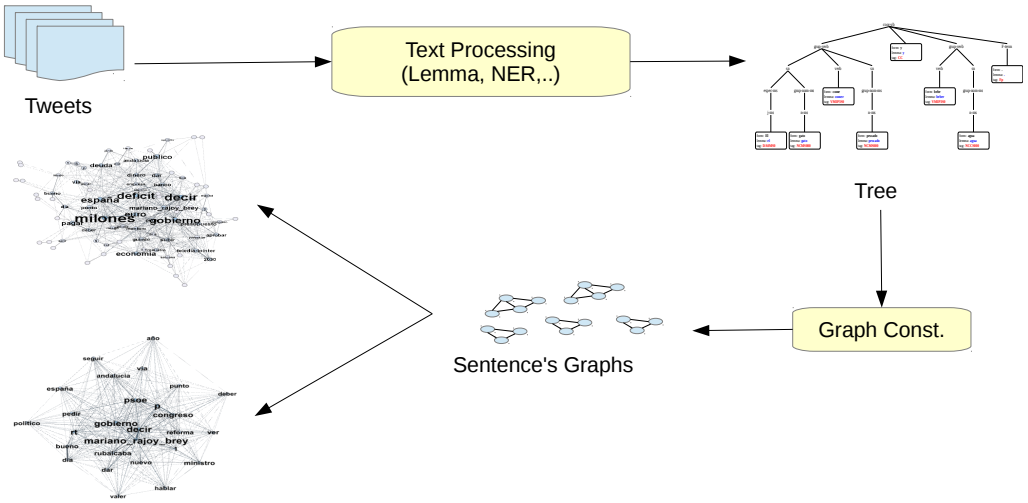

Figura 1: **Construcción del grafo de referencia.** A partir de un conjunto de textos pertenecientes a un mismo tema, se construye un grafo resultado de unir los grafos de cada sentencia encontrada en los tweets.

único término conjuntos de palabras que se refieren al mismo concepto (p.e. *Real*, *Real Madrid*, *Real Madrid C.F.*, etc.). Para ello y como prueba de concepto se ha utilizado un pequeño catálogo construido manualmente con algo menos de 100 nombres de entidades, con varias formas de nombrado para cada una. Para crear el catálogo, los textos de entrenamiento han sido inicialmente descompuestos en n-gramas sin límite de longitud, usando la técnica descrita en (Nagao y Mori, 1994). Tras la extracción de n-gramas estadísticamente significativos, el catálogo fue expandido manualmente tanto en conceptos similares (por ejemplo, nombres de medios de comunicación), como en diferentes formas de aparición de estos conceptos.

En el procesado de NER se ha realizado una búsqueda en el catálogo por cada aparición de n-grama en el texto para determinar si se refiere a una de las entidades catalogadas, en cuyo caso se sustituye por su nombre "canónico". Por ejemplo, el bigrama *Mariano Rajoy* se ha tratado internamente como una de estas entidades, en este caso la entidad de nombre canónico *mariano_rajoy_brey*. Todo este proceso se ha realizado de forma experimental, y creemos que una utilización más exhaustiva y un catálogo más completo podría mejorar la calidad de los resultados.

En resumen, el preprocesado de cada texto está compuesto de las siguientes fases, en este orden. Primero se eliminan las URL que contenga el texto. Segundo, se usa el diccionario SMS sustituyendo las abreviaturas y símbolos encontrados por su equivalente textual. En una tercera fase, se corrige la ortografía usando Hunspell. En una cuarta fase se identifica el idioma en que esta escrito el texto usando la *Language Detection Library* de Cybozy Labs (Shuyo, 2010), y se descarta si no está escrito en castellano. En la quinta fase se realiza el NER, sustituyendo las entidades encontradas por su nombre canónico. Hemos preparado el código para poder desactivar esta fase y así evaluar la efectividad de la misma. En la sexta fase se realiza la lematización de las palabras, usando Freeling. Finalmente, en la séptima y última fase del preprocesamiento se eliminan las palabras vacías (*stop words*).

3.2. Grafos de referencia

Para la construcción de los grafos de referencia se han utilizado dos técnicas que han producido resultados similares. Ambas utilizan grafos en los que los enlaces tienen pesos. La primera de las técnicas determina el peso de un enlace simplemente contando el número de veces en el que las dos palabras han aparecido conjuntamente en el mismo *tweet*. La segunda alternativa pondera ese peso mediante la distancia que hay entre las palabras. Dos palabras que aparecen juntas en el texto tienen pesos superiores a las palabras que aparecen en extremos opuestos de una fra-

se. Esta distancia se ha obtenido a partir del árbol proporcionado por Freeling en el análisis sintáctico. Para calcular la distancia de dos palabras se considera el número de saltos que deben darse en el árbol para llegar de una palabra a otra.

Otra variante que abrió nuevas alternativas en nuestro trabajo fue la decisión de incluir adicionalmente los sinónimos de las palabras. En esta variante, además de incluir las palabras o términos originales del texto, se añadían todos los sinónimos de éstas. Los pesos de los enlaces de estos sinónimos era algo menor.

En las pruebas realizadas, el uso de sinónimos disminuía la calidad de los resultados, posiblemente al usar principalmente medidas de centralidad que perdían capacidad de representación del tema del grafo.

También se probó el uso de sinónimos a la hora de aprovechar la información del grafo (no en su creación). En este caso no se detectaron mejoras significativas.

Pensamos que este tema debe ser abordado en un trabajo futuro ya que esperamos que este grafo aumentado pueda enriquecer los grafos de referencia y por lo tanto aumentar la tasa de categorización.

3.3. Categorización del texto

Finalmente, una de las principales cuestiones de nuestro enfoque está relacionado con problema de detección de similitud de grafos. La elección de la medida de similitud es una decisión compleja ya que existen numerosas medidas y no es evidente cuál puede resultar adecuada a nuestro problema. En nuestro trabajo hemos utilizado varias propuestas, pero todas ellas se basan en extraer los subgrafos resultado de filtrar los grafos de referencia por las palabras que aparecen en el mensaje a clasificar.

Es decir, para cada grafo de referencia se han extraído las palabras que aparecen en el mensaje a clasificar y se mantienen los enlaces que había entre ellas. Para cada tema, se obtiene así un subgrafo que puede ser incluso vacío si ninguna de las palabras del mensaje se encuentra dentro del grafo de referencia. El siguiente paso es determinar una o varias medidas topológicas que aplicadas a dichos subgrafos nos permitan determinar el tema o temas del mensaje.

Se han utilizado dos grandes tipos de medidas: las basadas en métricas de nodos y en las basadas en métricas de relaciones. Las métricas de nodo han sido principalmente dos: PageRank (Brin y Page, 1998) y HITS (Kleinberg, 1999). Para el cálculo de estas métricas se han utilizado las variantes correspondientes a grafos no dirigidos y con pesos en los enlaces. Una vez extraído el subgrafo, la decisión se basa en la suma de las medidas de los nodos (por ejemplo, la suma normalizada del PageRank para todos los nodos del subgrafo). Estas métricas demostraron ser muy útiles para determinar el tema del mensaje. Como se puede observar en el cuadro 1 no se han detectado grandes diferencias entre la utilización de PageRank o HITS. Dado que el método utilizado se basa en una suma de medidas, éste ha resultado muy útil por su sencillez y velocidad en la clasificación.

Aunque la construcción de los grafos de referencia y el posterior cálculo de PageRank puede ser un proceso laborioso, todavía necesitamos una serie de pasos adicionales. Dado que el tamaño de los grafos de referencia está muy influenciado por el conjunto de entrenamiento (p.e. número de *tweets* por cada tema), éstos no serán iguales, y por tanto hay que compensar este efecto por medio de una normalización de las medidas de centralidad. Bajo una primera hipótesis sencilla, en la que suponemos que a igualdad de representatividad los valores de las medidas de centralidad disminuyen según el número de nodos del grafo, hemos normalizado según el tamaño del grafo de referencia bajo consideración. Por otro lado, dado que estos valores también dependen de la topología del grafo de forma no predecible, hemos probado el uso de operaciones no lineales (en particular, potencias como 0,5 y 2), para aportar capacidad de representatividad al sistema.

La clasificación de un nuevo mensaje a partir del texto preprocesado es rápida. Como primera aproximación es suficiente sumar las métricas de centralidad de cada palabra por tema, y optar por aquél que ofrece un mayor valor. Con este método se obtienen valores de aproximadamente un 60 % de acierto. No obstante, el uso de clasificadores más sofisticados consigue mejores resultados, como se mostrará más adelante.

Nuestro trabajo también ha abordado las métricas de enlaces. Dado que cada enlace tiene un peso, es posible calcular métricas usando estos valores. Se han intentado varias técnicas, pero todas ellas están basadas

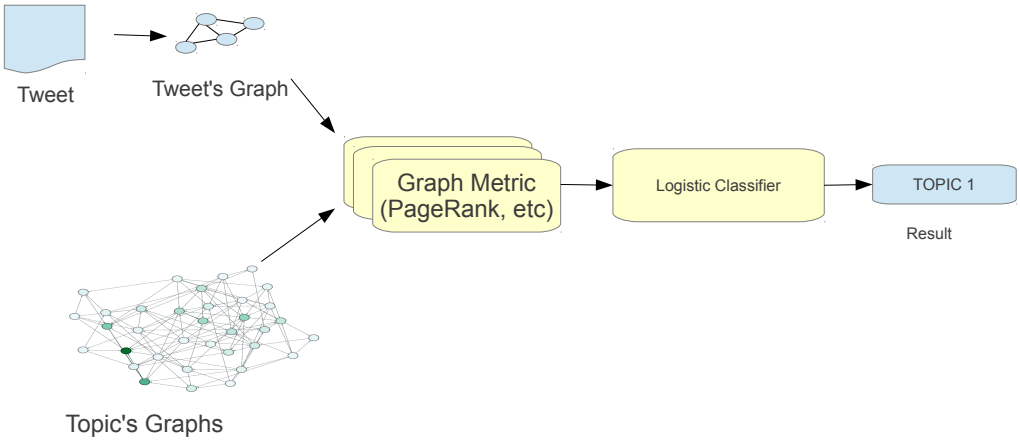

Figura 2: **Clasificación del texto.** Cuando se desea clasificar un texto se buscan similitudes entre el grafo que representa el texto y los grafos de referencia. Para ello se utilizan técnicas de similitud de grafos.

en la densidad del subgrafo resultante (suma ponderada de los pesos de los enlaces). Esta técnica no ha obtenido resultados mucho mejores por sí misma, pero durante la evaluación sobre el conjunto de entrenamiento ha resultado ser clave cuando se combina con las descritas anteriormente. No obstante, en la evaluación sobre el conjunto de test la mejora ha sido un poco más pequeña.

Posteriormente hemos evaluado el uso de clasificadores, en particular las implementaciones de Weka (Hall et al., 2009). Cada *tweet* estaba representado por un vector formado por todas las métricas disponibles (suma de PageRank, suma de HITS, densidad del grafo, etc.) y para cada grafo de referencia. En total disponemos de un vector formado por 70 valores numéricos. De todos los métodos de clasificación disponibles en Weka, determinamos que la familia de *Logistic* era la que producía tasa de aciertos más alta. En concreto el método de *MultiClass Classifier* aplicado a la familia *Logistic* resultó ser el que mejores resultados produjo sobre el conjunto de entrenamiento de forma consistente.

4. Resultados y discusión

Para mostrar el funcionamiento hemos evaluado nuestro sistema configurado de distintas formas. En todas las ejecuciones hemos entrenado Weka con el 100 % del conjunto de entrenamiento (formado por unos 7.000 *tweets*) y hemos evaluado el modelo resultante con algo menos de 60.000 *tweets* del conjunto proporcionado para la evaluación tras la aparición de los resultados del concurso de 2013, descartando de los resultados algunos *tweets* de los que no disponemos. El algoritmo de Weka usado ha sido siempre el *SimpleLogistic* descrito arriba.

En el cuadro 1 se muestran los resultados obtenidos en las distintas pruebas realizadas. En la columna etiquetada como *Configuración* se indican los atributos del texto que se han utilizado: PageRank (PR), HITS, densidad de grafo (GD), y las modificaciones aplicadas. Estos atributos se han generado para cada *tweet* tanto durante el entrenamiento como en la evaluación.

En la columna *NER* se indica si se ha realizado la fase NER en el preprocesamiento de los *tweets* o no. Como dijimos, hemos desactivado esta fase en algunas ejecuciones para evaluar su influencia en los resultados. En la columna *Acierto* indicamos el porcentaje de *tweets* a los que el sistema ha asignado un tema de los que se consideran correctos según los datos de evaluación suministrados.

Los cuadros 2 y 3 recogen información acerca de la distribución de las categorías tanto de los *tweets* de entrada como de los resultados del categorizador usado en el experimento 1. Hemos expresado los resultados por categoría de esta forma, ya que la posibilidad de múltiples categorías por *tweet* provoca que una matriz de confusión completa fuese muy

Experimento	Configuración	NER	Acierto (%)
1	$PR^{0,5}$, PR, PR^2, $HITS^{0,5}$, HITS, $HITS^2$, GD	Sí	71,90
2	$PR^{0,5}$, PR, PR^2, $HITS^{0,5}$, HITS, $HITS^2$	Sí	71,05
3	$PR^{0,5}$, PR, PR^2, $HITS^{0,5}$, HITS, $HITS^2$	No	66,92
4	PR	Sí	68,66
5	$PR^{0,5}$	No	67,84
6	$PR^{0,5}$	Sí	70,86
7	$PR^{1/3}$	Sí	70,75
8	$PR^{0,1}$	Sí	68,90
9	$HITS^{0,5}$	Sí	71,32
10	$HITS^{1/3}$	Sí	71,35

Cuadro 1: Comparativa de Resultados

grande y poco intuitiva. De esta manera, en el cuadro 3 la tasa de acierto se interpreta como la proporción de las predicciones etiquetadas dentro de la categoría cuyo *tweet* pertenece, por lo menos, a esa categoría.

A partir de los resultados se puede apreciar que el uso de una métrica de centralidad sobre la otra no ofrece diferencias demasiado significativas, incluso cambiando el tipo de normalización empleado. Por el contrario, el uso de una normalización específica sí aporta una mejora sensible (en torno al 2 %, por ejemplo, entre los experimentos 4 y 6). Junto a los buenos resultados obtenidos por el uso de métricas de centralidad, esto hace pensar que la elección de una normalización adecuada es primordial para la mejora de los resultados. O en todo caso, el uso de una métrica que sea capaz de tener todos los factores (tamaño, topología, etc.) en cuenta. Por tanto, es un campo importante a investigar en el futuro.

También hemos apreciado durante la ejecución de los experimentos la sensibilidad al vocabulario disponible, sobre todo por grafo: temas con muy pocos *tweets* tendían a ser ignorados, posiblemente por la poca representatividad de los grafos generados, por ejemplo en el caso de *tecnología*. Un posible trabajo consistiría en evaluar la sensibilidad con grupos de entrenamiento mayores y así poder determinar la importancia de este efecto.

Asimismo esta sensibilidad podría probarse incrementando el tamaño del diccionario NER, para asegurarnos de que consigue representar con más detalles los temas en los que el sistema clasifica. A pesar de contar con un diccionario pequeño su influencia es importante en los resultados. Por ejemplo, entre los experimentos 2 y 3 se aprecia una mejora de un 4 %, o un 3 % entre los experimentos 5 y 6. Por tanto puede merecer la pena invertir esfuerzos en este aspecto.

Por otra parte, la evaluación automática de los modelos predictivos en Weka está limitada ya que ésta no es capaz de considerar más de una predicción por vector. Posiblemente un sistema automatizado de evaluación propio permitiría conseguir mejores resultados, ya que el API de Weka sí ofrece el vector de resultados por categoría, y así poder ajustar el sistema de forma más próxima al problema.

Sobre los resultados por categoría recogidos en el cuadro 3, cabe comentar que el sistema tiene un funcionamiento muy sesgado hacia las categorías principales (*política* y *otros*) que acaparan el 46.4 % y el 49.5 % de los *tweets* originales. En estos casos el sistema consigue unos resultados en torno al 78 %. Sin embargo, el resto de categorías presenta un comportamiento más pobre, muchas ellas por debajo del 50 %. Particularmente interesante es el caso de *entretenimiento*, ya que siendo la tercera categoría en frecuencia de aparición, la tasa de acierto es del 38 %.

Posiblemente un experimento adicional con un entrenamiento más proporcionado pueda distinguir si este comportamiento es debido a un entrenamiento desigual o a causas intrínsecas al sistema. No obstante, dado el número limitado de textos de entrenamiento, por ejemplo en la categoría *literatura*, hace pensar que será necesario un conjunto de entrenamiento más completo que el presente.

Categoría	Tweets	Aparición (%)	Aparición normalizada (%)
cine	596	1.0	0.9
deportes	135	0.2	0.2
economía	2549	4.2	3.7
entretenimiento	5421	8.9	7.8
fútbol	823	1.4	1.2
literatura	93	0.2	0.2
música	1498	2.5	2.1
otros	28191	46.4	40.5
política	30067	49.5	43.2
tecnología	287	0.5	0.4

Cuadro 2: Tweets por categoría

Categoría	Predicciones	Proporción sobre total	Tasa de acierto
cine	460	0.77	43.26
deportes	67	0.11	47.76
economía	612	1.03	50.16
entretenimiento	6919	11.66	38.98
fútbol	420	0.71	52.62
literatura	60	0.10	25.00
música	1095	1.84	51.60
otros	19753	33.29	77.00
política	29890	50.38	78.27
tecnología	58	0.10	32.76

Cuadro 3: Resultados por categoría

Referencias

Brin, Sergey y Lawrence Page. 1998. The anatomy of a large-scale hypertextual web search engine. *Comput. Netw. ISDN Syst.*, 30(1-7):107–117, Abril.

Fernández Anta, Antonio, Luis Núñez Chiroque, Philippe Morere, y Agustín Santos. 2013. Sentiment analysis and topic detection of Spanish tweets: A comparative study of of NLP techniques. *Procesamiento del Lenguaje Natural*, 50:45–52.

Hall, Mark, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, y Ian H. Witten. 2009. The WEKA data mining software: an update. *SIGKDD Explorations*, 11(1):10–18.

Kleinberg, Jon M. 1999. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, Septiembre.

Nagao, Makoto y Shinsuke Mori. 1994. A new method of n-gram statistics for large number of n and automatic extraction of words and phrases from large text data of Japanese. En *Proceedings of the 15th conference on Computational Linguistics, COLING 1994, Volume 1*, páginas 611–615. Association for Computational Linguistics.

Padró, Lluís, Samuel Reese, Eneko Agirre, y Aitor Soroa. 2010. Semantic services in freeling 2.1: Wordnet and ukb. En Pushpak Bhattacharyya Christiane Fellbaum, y Piek Vossen, editores, *Principles, Construction, and Application of Multilingual Wordnets*, páginas 99–105, Mumbai, India, February. Global Wordnet Conference 2010, Narosa Publishing House.

Shuyo, Nakatani. 2010. Language detection library for java. http://code.google.com/p/language-detection/.