FACULTADE DE INFORMÁTICA

Trabajo Fin de Máster Universitario en Ingeniería Informática

Análisis de contenidos en Twitter: clasificación de mensajes e identificación de la tendencia política de los usuarios

Autor:

David Vilares Calvo

Directores:

Miguel A. Alonso Pardo

Carlos Gómez Rodríguez

Junio de 2014

Índice general

Índice de figuras

Índice de tablas

Resumen

Las millones de opiniones y críticas expresadas en Twitter cada día están empezando a constituir una importante fuente de información para numerosas compañías y organizaciones, que ven en este medio un lugar para sondear su área de influencia. Conocer la percepción sobre productos, servicios, eventos o personalidades relevantes, así como monitorizar su reputación online son algunos de los objetivos que las compañías se han marcado a corto plazo. Uno de los primeros problemas a los que se enfrentan estas empresas es discriminar los mensajes pertenecientes a su ámbito de negocio en un medio tan ruidoso como Twitter, donde es posible encontrar opiniones sobre prácticamente cualquier tema. A este respecto, las funcionalidades de búsqueda de Twitter se limitan a sencillas funciones como búsqueda por palabras clave, capacidades de búsqueda por idioma o recuperación de los tuits de un determinado autor. Este trabajo fin de máster aborda el desarrollo de nuevas técnicas que permitan identificar los temas sobre los que se habla en un tuit. El objetivo es proporcionar capacidades para recuperar opiniones y críticas relacionadas con un determinado ámbito o sector de negocio, filtrando así los tuits no relacionados. La tarea se aborda desde una perspectiva de procesamiento del lenguaje natural. En primer lugar se lleva a cabo un preprocesado *ad-hoc* de los textos, para luego analizar morfosintácticamente los mismos, obteniendo su estructura sintáctica, la cual sirve de punto de partida para el posterior análisis semántico. El problema es abordado desde un punto de vista de clasificación multi-etiqueta, dado que no es extraño que en un mismo mensaje un usuario relacione diferentes temas. Los resultados experimentales demuestran la validez de las propuestas, que mejoran el estado del arte en la clasificación de tópicos de mensajes escritos en castellano.

En lo referido a la identificación de tendencias políticas, partiendo de un subconjunto de los modelos propuestos para la clasificación de tópicos se construyen clasificadores capaces de diferenciar entre usuarios conservadores, progresistas y de centro. La novedad de la tarea,

junto a su alto nivel de abstracción y subjetividad, la convierten en un reto interesante y complejo dentro del ámbito del análisis automático de textos, algo que queda reflejado en los resultados experimentales.

Palabras clave

- *Contenido del TFM:* procesamiento del lenguaje natural, aprendizaje automático, clasificación de tópicos, identificación de tendencia política.
- *Herramientas utilizadas:* NLTK, Maltparser, Twitter API, WEKA, Python.

Capítulo 1

Introducción

Con la aparición de la Web 2.0 y el auge de los medios sociales, el análisis de textos web se ha convertido en un tema de gran interés en muchos ámbitos profesionales. Las redes sociales actúan como un punto de encuentro donde millones de usuarios pueden publicar y compartir opiniones, información o simplemente trivialidades sobre todo tipo de temas. Poder analizar y comprender toda esa información pública se está convirtiendo en uno de los principales objetivos por parte de empresas y organizaciones, que ven en estos lugares una fuente de información para analizar que se dice sobre su área de influencia. A este respecto, conocer la percepción de los usuarios sobre un determinado producto, servicio o evento, así como realizar estudios de mercado sobre los usuarios que publican sus pareceres en la web se han convertido en alguno de sus objetivos a corto plazo. En la última década, los esfuerzos se habían centrado especialmente en la monitorización de textos largos publicados en foros o blogs. Sin embargo, dado el reciente éxito de redes de microblogging como MySpace, Facebook o Twitter; las organizaciones han empezado a centrar su interés en el análisis de textos cortos. En concreto, este trabajo fin de máster se centra en tareas de análisis de mensajes en Twitter, red social que ha alcanzado una gran popularidad y donde los usuarios publican sus experiencias en mensajes de hasta 140 caracteres, popularmente conocidos como *tuits*.

Uno de los problemas a los que se enfrentan las organizaciones a la hora de monitorizar Twitter es la gran cantidad de tuits no relacionados con los intereses de la organización. Actualmente, en torno a 500 millones de mensajes son publicados por unos 100 millones de usuarios activos, lo que convierte a esta red social en un medio ruidoso [41]. En este contexto, es necesario aplicar previamente filtros para tratar de eliminar la mayor cantidad de mensajes no relacionados con el ámbito en cuestión y realizar un análisis fiable. A este respecto, las funcionalidades de búsqueda de Twitter pueden ser útiles a la hora de encontrar

mensajes relacionados con un aspecto muy específico, como puede ser un modelo de coche particular o un actor, pero no son prácticas cuando se pretenden recuperar tuits sobre un tema más general. Por ejemplo, si se desean recuperar tuits relacionados con una película, y la consulta de búsqueda es el propio nombre del filme, es probable que muchos de las opiniones recuperadas no traten realmente sobre cine, dado que muchas de las palabras del título harán referencia a otros dominios.

Para resolver este problema, este trabajo fin de máster propone una aproximación de clasificación por temas para mensajes Twitter. El sistema permite asociar un tuit a más de una categoría y así detectar cómo los usuarios relacionan distintas categorías, tales como *economía* y *política* o *cine* y *tecnología*.

El enfoque lingüístico es aprovechado entonces para evaluar otras tareas de análisis automático de texto. En concreto, se aborda la identificación de tendencias políticas en Twitter para los usuarios que la manifiestan, distinguiendo tres clases: *derecha*, *izquierda* y *centro*. Dada la falta de corpus de entrenamiento existentes para el castellano ha sido necesario construir uno de manera semi-automática. Los resultados experimentales confirman la complejidad de esta tarea, dado su alto nivel de abstracción y subjetividad.

1.1. Objetivos

La detección y clasificación de las temáticas que tratan los textos web se ha convertido en un aspecto que ha despertado el interés de diversas empresas. Sin embargo, la mayor parte de los sistemas existentes se centran en el análisis de textos largos y generalmente en lengua inglesa. Además, la literatura relacionada con la monitorización de temáticas en redes sociales de microblogging, como es el caso de Twitter, es escasa. En relación al castellano, la aparición del Taller de Análisis del Sentimiento en la SEPLN [44], conocido como TASS 2013 ha supuesto un buen punto de encuentro donde los profesionales ponen a prueba sus sistemas en un marco estándar, que permita evaluar las distintas aproximaciones bajo las mismas condiciones. Estos sistemas suelen seguir un enfoque puramente léxico [23, 32] o bien basarse en grafos [14], centrándose en un único enfoque, sin tratar de combinar diferentes perspectivas. Además muchos de estos sistemas solo permiten asignar un mensaje a una categoría, cuando no es extraño que los usuarios relacionen distintas temáticas en un mismo mensaje. El sistema aquí propuesto pretende cubrir estas carencias, combinando información léxica, sintáctica, psicométrica y semántica para tratar de mejorar el rendimiento de los

sistemas actuales. Se presentan diferentes enfoques iniciales para tratar de medir su potencial en un entorno de clasificación múltiple.

Este trabajo también aborda el problema de la clasificación de perfiles de usuarios web. En concreto, se pretende categorizar a los usuarios de Twitter en función de su tendencia política, a partir de tuits. Las capacidades de un sistema de identificación de tópicos permiten seleccionar los usuarios que suelen opinar sobre temas políticos, identificando, por tanto, a aquellos que son proclives a mostrar públicamente una ideología política. Sobre este aspecto no existe mucho trabajo relacionado, incluso para el inglés [15, 27]. En relación al castellano, de nuevo el TASS 2013 ha supuesto un marco donde los profesionales pueden evaluar sus aproximaciones en este ámbito, para usuarios que habitualmente expresan sus opiniones en castellano.

1.2. Metodología

La metodología escogida ha sido una iterativa incremental, desarrollando el núcleo del sistema además de un primer incremento.

- *Análisis, diseño e implementación del sistema de procesado del lenguaje natural.*
- *Análisis, diseño, implementación y pruebas de los módulos de clasificación y extracción de características.*

El sistema obtenido se emplea para la evaluación de las dos tareas propuestas en este proyecto a través de los corpora disponibles:

- *Experimentación para la clasificación de tópicos.*
- *Experimentación para la identificación de tendencias políticas.*

1.3. Planificación temporal

Este trabajo fin de máster se ha desarrollado de acuerdo a la siguiente planificación temporal, con fecha de inicio a 16 de diciembre del año 2013; momento en el que se comenzó a estudiar el trabajo relacionado con las áreas propuestas. El análisis, diseño e implementación de estas actividades fue responsabilidad del alumno que ha representado el rol de *Analista-Diseñador-Programador*, con una dedicación de 20 horas semanales. Por su parte, los dos codirectores fueron los encargados de supervisar el trabajo, actuando como *Jefe de proyecto*,

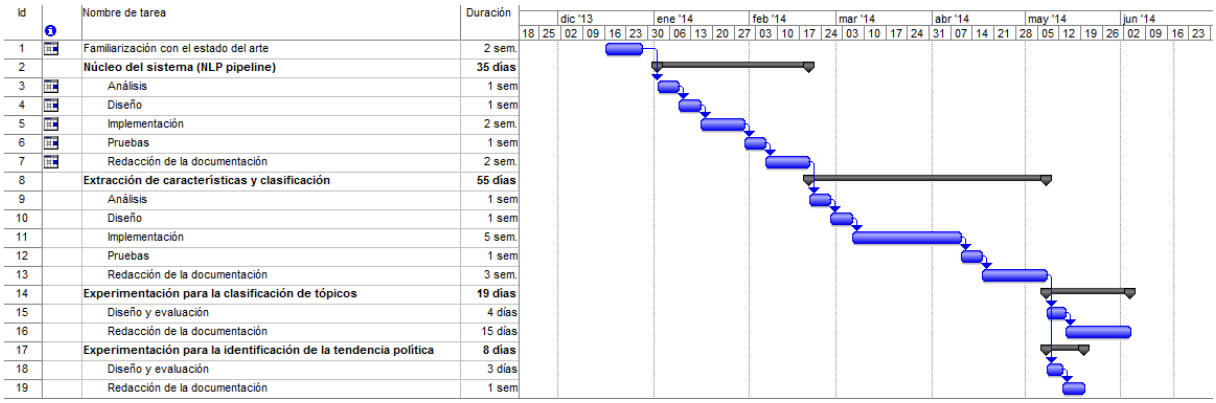

Figura 1.1: Diagrama de Gantt con la planificación del proyecto

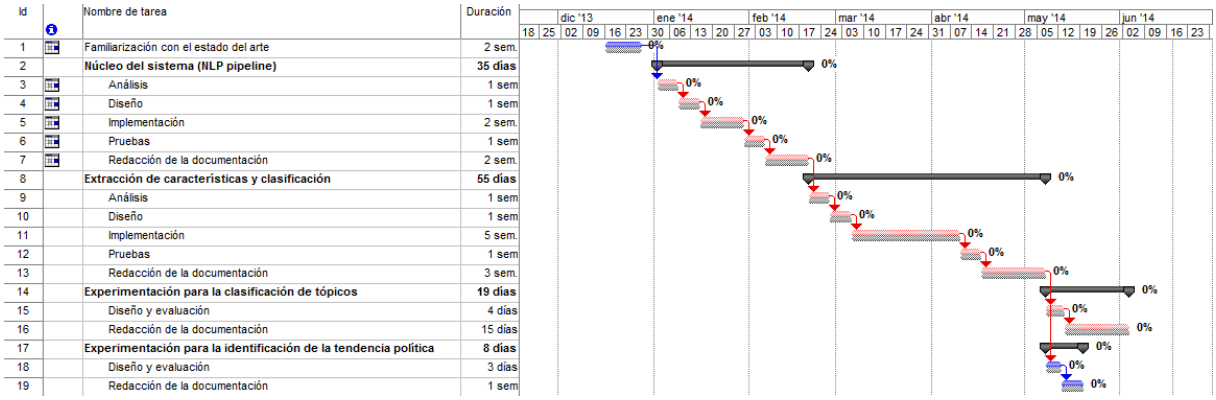

Figura 1.2: Línea de base del proyecto de este TFM

participando en las fases de análisis para cada una de las etapas. La figura 1.1 ilustra la planificación de este TFM mediante un diagrama de Gantt.

El siguiente paso fue establecer la línea de base. La mayor parte del proyecto se realiza de manera secuencial, con actividades relacionadas en modo *Comienzo-Fin*. No obstante, las etapas finales de experimentación para la detección de tópicos y tendencias políticas, podrían llevarse a cabo de forma paralela. Ello incurre en el que camino crítico viene marcado por la duración de la más larga que en este caso se corresponde con la experimentación para la clasificación de tópicos, como queda marcado en la línea de base ilustrada en la figura 1.2.

Por último, la figura 1.3 refleja el diagrama de Gantt correspondiente al seguimiento al 100 %, dando por finalizado el proyecto el 12 de junio de 2014. Además de desviaciones en el plan inicial, las fases de experimentación se llevaron a cabo secuencialmente, dada la falta de disponibilidad de recursos.

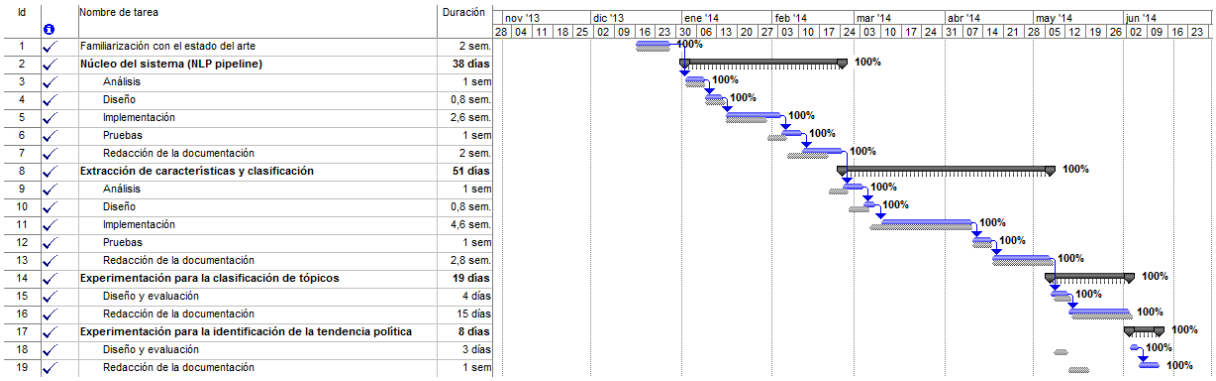

Figura 1.3: Diagrama de Gantt para el seguimiento al 100 %

1.4. Estructura

La estructura de este trabajo se organiza como sigue. En el capítulo 2 se presenta el trabajo existente relacionado con las tareas propuestas en el trabajo (secciones 2.1 y 2.2, respectivamente). A continuación se presentan la herramientas y recursos empleados en este trabajo fin de máster, lo que ocupa el capítulo 3. Por su parte, el capítulo 4 describe el procesamiento del lenguaje natural aplicado sobre los textos web como paso previo a su clasificación. Se describen las etapas de preprocesado, segmentación de oraciones y palabras, así como de análisis morfológico y sintáctico. El capítulo 5 presenta la aproximación implementada para resolver la primera de las dos tareas resueltas en este trabajo fin de máster: la identificación de las temáticas que trata un tuit. Se detallan los distintos enfoques propuestos, describiendo sus principales componentes y características. La identificación de las tendencias políticas de usuarios es descrita y abordada en el capítulo 6. En el capítulo 7 se ilustran y discuten los resultados para ambas tareas. Por último, las conclusiones y el trabajo futuro son presentadas en en el capítulo 8. El Apéndice A ilustra la distribución de frecuencias de corpus utilizado para entrenar y evaluar el clasificador de tópicos.

Capítulo 2

Análisis de textos web

Este capítulo presenta el estado del arte sobre las dos tareas tratadas en este trabajo fin de máster: la clasificación de temáticas en mensajes de Twitter y la detección de las tendencias políticas de los usuarios de esta misma red social.

Un enfoque tradicional a la hora de resolver tareas de análisis automático de textos se basa en la creación de sistemas de conocimiento, en los cuales se aplican una serie de reglas definidas manualmente por un conjunto de expertos, para establecer criterios de clasificación. Sin embargo, con la aparición de las técnicas de aprendizaje automático, la clasificación de textos ha empezado a enfocarse también desde esta perspectiva: dado un conjunto de documentos, estos son clasificados en diversas categorías según las necesidades de la aplicación y el usuario. A partir de ahí, se extraen una serie de *características*, que representen una abstracción de este texto [9, capítulo 10], con el fin de entrenar un clasificador supervisado que aprenda a diferenciar entre las distintas clases. Este enfoque presenta además varias ventajas respecto a la perspectiva basada en conocimiento, entre las que destacan: un rendimiento robusto, la facilidad de adaptación a diferentes dominios y ahorro en términos de tiempo y costes en lo referido a las tareas de los expertos [35].

La clasificación supervisada de textos web ha sido tradicionalmente utilizada para la categorización de textos largos, típicamente extraídos de foros o blogs. Tareas como determinar si un documento se refiere o no a una entidad [16, 46], si en él se expresa o no una opinión [42] o detectar cuál es la temática que se trata en el mensaje son algunas de las tareas en las que se han invertido esfuerzos por parte de la academia y la industria. Sin embargo, el reciente éxito de las redes sociales de microblogging, como MySpace, Facebook o más recientemente Twiter, ha despertado el interés de diversas organizaciones por analizar este tipo de textos.

A continuación se describe el trabajo relacionado con las dos tareas propuestas, centrándo-

nos especialmente en las técnicas disponibles para Twitter desarrolladas para el castellano, idioma tratado en este trabajo fin de máster.

2.1. Clasificación de tópicos en microtextos

Twitter se ha convertido en una red social donde los usuarios pueden publicar y encontrar información sobre prácticamente cualquier tema, lo que convierte a este medio en una gran red de información. Ello ha provocado que los creadores de contenidos estén empezando a mostrar interés en identificar las temáticas de todos esos mensajes, para así filtrar los tuits que hacen referencia a un determinado producto, servicio o evento. En [3] se presentan una lista de 12 tópicos que habitualmente son discutidos en esta red social: *política*, *altruismo*, *eventos*, *tecnología*, *juegos*, *idiomas*, *música*, *personalidad*, *películas*, *celebridades*, *estilo de vida* y *deportes*. Por su parte Sriram *et al.* [36] diferencian cinco categorías: *noticias*, *eventos*, *opiniones*, *ofertas* y *mensajes personales*. En su estudio proponen un sistema de clasificación que combina un modelo de bolsa de palabras con siete características de valor binario, extraídas del contenido de los mensajes: palabras abreviadas, términos con opinión, utilización de fenómenos para enfatizar términos, empleo de argot, frases que hagan referencia a eventos temporales o la existencia de menciones a un usuario (precedidos en esta red social del símbolo @), bien al principio, o dentro de un tuit. Los resultados experimentales, sobre un corpus de 5 400 tuits y 684 autores, permitieron extraer conclusiones interesantes, como la relevancia de la característica que hace referencia a la aparición del nombre del autor a la hora de diferenciar las temáticas. En la misma línea, Thongusk *et al.* [39] utilizan el *Latent Dirichlet Allocation* (LDA) para agrupar un conjunto de términos extraídos de una colección de tuits en un conjunto de 50 temáticas, que son posteriormente utilizados para entrenar un clasificador basado en Support Vector Machine (SVM) . Los resultados sugieren que esta técnica mejora de manera estadísticamente significativa el rendimiento obtenido por un clasificador basado en una bolsa de palabras.

Por su parte, Fiaidhi *et al.* exploran si distintos enfoques de clasificación pueden mostrar variaciones en su rendimiento para categorizar distintas temáticas. Concretamente, entrenan 4 clasificadores distintos para distinguir entre doce clases (*política, educación, salud, marketing, música, noticia&medios, entretenimiento&deportes, ordenadores&tecnología, mascotas, comida, familia y otros*). La novedad de su propuesta reside en establecer un método para determinar cuál es el mejor clasificador para cada tópico. Para evaluar su propuesta utilizaron

un corpus de 100 000 tuits, obteniendo una precisión de un 75 %.

Gattani *et al.* [19] abordan y describen un sistema capaz de trabajar en tiempo real, algo obligado cuando se desea aplicar una técnica de clasificación de tópicos en un entorno empresarial. Entre otras funcionalidades, su aproximación soporta extracción de entidades, clasificación y etiquetación. En lo referido a la identificación de temáticas, los autores definen un conjunto de 23 tópicos a partir de una base de conocimiento extraída de Wikipedia. Para la evaluación de su propuesta emplearon un reducido conjunto de test, obteniendo un 50 % de precisión.

Por otro lado, una de las peculiaridades de Twitter reside en los *hashtag*, palabras clave precedidas del símbolo *'#'*, que sirven para etiquetar un mensaje y que con frecuencia representan un evento concreto. Lee *et al.* [21] presentan en su trabajo una técnica para identificar las temáticas de los hashtags más populares, conocidos como *trending topics*, diferenciando hasta un total de 18 categorías: *arte&diseño*, *libros, negocios, moda, comida&bebidas, salud, vacaciones&citas, humor, música, política, religión, ciencia, deportes, tecnología, tv&películas, otras noticias, otros*. Para ello, dado un hasthag, lo autores recopilan una colección de tuits que lo contengan, agrupándolos en un único documento, sobre el que finalmente se identifica el tópico. Para evaluar su enfoque, emplearon dos técnicas distintas de clasificación: (1) un modelo basado en una bolsa de palabras y (2) un árbol de decisión, obteniendo una precisión del 65 % y 70 %, respectivamente, sobre un corpus de 768 trending topics.

2.1.1. Clasificación en mensajes de Twitter escritos en castellano

El trabajo relacionado descrito sobre estas líneas muestra distintas aproximaciones que consideran un número diferente de tópicos y que han sido evaluadas sobre colecciones distintas de documentos, dificultando una comparación entre ellas en términos de rendimiento. En lo referido al castellano, afortunadamente existe un marco de evaluación estándar que permite evaluar el rendimiento de distintos sistemas de clasificación de tópicos bajo las mismas condiciones y criterios: el corpus TASS 2013 [45]. Se trata de una colección de tuits escritos por distintas personalidades públicas que se encuentran anotados con las temáticas que en ellos se tratan. Son diez los tópicos considerados: *política, entretenimiento, economía, música, fútbol, tecnología, deportes* (distintos de fútbol), *literatura* y una categoría adicional *otros*, para representar los textos que pertenecen a otros temas. Distintos autores han evaluado sus técnicas y aproximaciones sobre este corpus, convirtiendo a esta colección en el estándar *de facto* para la evaluación de técnicas de identificación de temáticas en tuits escritos en español.

Batista y Ribeiro [4] proponen un sistema de clasificación basado en modelos de regresión logística. Mediante clasificadores de máxima entropía para eventos independientes, crean un clasificador binario para cada tópico que estima la probabilidad de que un tuit pertenezca o no a dicha categoría. Estos clasificadores toman como atributos de entrada los unigramas y bigramas de palabras para cada mensaje, así como ciertas características propias de la jerga de Twitter, seleccionando el tópico más probable en función del valor de confianza obtenido para cada clasificador. Existen otros enfoques similares, como el propuesto por Pla y Hurtado [32]: en su estudio presentan una cascada de clasificadores binarios SMO [33], pertenecientes a la familia de los SVM (Support Vector Machine), que son entrenados para cada tópico y posteriomente utilizados para clasificar un conjunto de tuits desconocidos. La cascada de clasificadores asigna a cada tuit las temáticas detectadas por al menos un clasificador. Dado que cabe la posibilidad de que ningún tópico sea predicho, los autores emplean una segunda cascada de clasificadores binarios libSVM [12], a modo de *back-off*. Como resultado, cada clasificador devuelve un factor de probabilidad que indica la confianza con la que un tuit puede ser asignado a una categoría. En este caso, los autores seleccionan el tópico para el que se obtiene un mayor factor confianza, descartando todos los demás.

La efectividad de los clasificadores SVM ha sido estudiada por otros autores, sin considerar una arquitectura en cascada. Martínez-Cámara *et al.* [23] emplean un clasificador SVM entrenado con una bolsa de términos constituida por palabras extraídas del corpus, así como una colección de hashtags y palabras extraída de Google AdWords KeyWordTool.[1]

Por su parte Martín-Wanton y Carrillo de Albornoz [22] construyen una colección de palabras clave para describir a los distintos tópicos, en términos de divergencia Kullback-Leibler (KLD) [9, capítulo 9]. Es decir, cada tópico es definido como una lista ordenada de términos clave, utilizada para identificar las temáticas del tuit. Además definen un conjunto de eventos de acuerdo con un modelo LDA. Para determinar a que tópicos pertenece cada ejemplo, se obtiene la temática con una correlación mayor, comparando la clasificación de palabras de cada tópico con la clasificación de la palabras más probables para un evento dado.

El KLD ha sido utilizado también por otros autores dentro de sus enfoques. Castellano *et al.* lo emplean dentro de su perspectiva basada en Recuperación de Información. En particular, en [10] presentan su propuesta original, donde el conjunto de tuits de entrenamiento es representado e indexado mediante modelos de lenguaje. A continuación, para determinar

[1] http://adwords.google.com/o/KeywordTool reemplazada por Google Keyword Planner para los usuarios registrados.

la categoría de un nuevo tuit, se utiliza su contenido como una consulta contra el índice en el que se encuentran los tuits almacenados previamente. En [11] expanden su propuesta, donde analizan como distinta información presente en los mensajes de Twitter puede ser utilizada en la clasificación de temáticas. En su estudio obtienen conclusiones interesantes, como el reducido impacto del nombre de entidades en tareas de clasificación, lo que refuerza la necesidad de indexar todas las palabras del contenido de los tuits para poder explotar adecuadamente un enfoque basado en recuperación de información.

Los enfoques basados en recuperación de información también han sido tenidos en cuenta por otros autores. Montejo-Ráez *et al.* [26] preprocesan el contenido de los tuits para luego convertirlo en un vector de datos, donde cada característica está ponderada de acuerdo a un esquema de pesado tf-idf [9, capítulo 2]. Dicha información se emplea entonces para construir una matriz términos-temáticas que ayuda a clasificar los tópicos de un tuit. La aproximación no obtuvo un buen rendimiento, lo cual fue atribuido al reducido tamaño del conjunto de entrenamiento.

Un enfoque alternativo es el propuesto por Cordobés *et al.* en [14], los cuales presentan una técnica basada en similitud de grafos para identificar las temáticas que trata un tuit. En este enfoque, cada palabra constituye un vértice del grafo. Una conexión entre dos vértices (arco) representa que esos dos elementos aparecen conjuntamente en algún tuit. A cada arco se le asigna un peso que representa la frecuencia de aparición conjunta de ambos términos. Para reducir la dispersión, las palabras son normalizadas a su raíz gramatical. El conjunto de entrenamiento se emplea para construir un grafo para cada tema, uniendo los grafos obtenidos para los tuits de esa categoría. Después, se construye un grafo para cada tuit del conjunto de test y se busca cuál de los grafos de referencia es más similar, mediante técnicas basadas en [7]. Como consecuencia, sólo una categoría puede ser asignada a un tópico, es decir, no permite realizar clasificación multi-etiqueta. Los autores presentan conclusiones interesantes, como que la utilización de sinónimos disminuye el rendimiento, en contra de lo esperado.

2.2. Clasificación de la tendencia política

La clasificación de la tendencia política de usuarios es una tarea muy reciente, por lo que la literatura relacionada con este tema es escasa. Cabe destacar el estudio de Dalvean [15] donde se plantean dos retos. Por un lado, determinar cómo los discursos de los parlamentarios de Australia de las dos grandes facciones del país, el *Australian Labor Party* y la *Liberal*

National Party Coalition, pueden ser clasificados o no dentro de sus respectivos grupos. Para ello emplea técnicas de aprendizaje automático y de análisis de textos, obteniendo una precisión del 73 %, lo que sugiere que el discurso de los políticos australianos es un aspecto fiable para identificar su ideología. Los factores de confianza proporcionados por el clasificador utilizado, son tomados como punto de partida para abordar el segundo reto planteado: situar a los parlamentarios de estos dos partidos en una horquilla ideológica. El autor pretende así identificar a los políticos más moderados y radicales en ambos partidos.

Por su parte Mullen y Malouf [27] presentan un prototipo de análisis del sentimiento aplicado sobre discursos políticos. El análisis del sentimiento es una reciente área de investigación centrada en el análisis automático de la información subjetiva. Una de sus subtareas más populares es la clasificación de la polaridad, centrada en determinar si un texto es positivo, negativo o mixto. Los autores parten de estas bases para desarrollar técnicas que indiquen si una respuesta a un mensaje en un determinado hilo ilustra una crítica o un punto de vista opuesto al mensaje original. Los resultados experimentales hacen concluir a los autores que las técnicas tradicionales de análisis del sentimiento no son adecuadas para detectar discrepancias en foros. Ello es achacado a la dificultad del dominio, muy abstracto, lo que complica la tarea, algo ya comentado por otros autores [40]. Además, plantean desarrollar nuevos métodos que tengan en cuenta la forma en que los usuarios interactúan.

En lo referido al castellano, el corpus TASS también dispone de una colección de usuarios de Twitter, anotados hasta con cuatro tendencias políticas distintas. La tarea se encuentra todavía en una etapa muy inicial, tanto a nivel de definición como de propia implementación, aunque existen algunos enfoques que ya han sido evaluados sobre el corpus TASS. García y Thelwall [18] presentan una técnica en dos fases. En primer lugar presentan un método basado en lexicones para identificar los tuits que tratan sobre política. A continuación, sobre el conjunto de tuits filtrados, los autores adaptan el algoritmo SentiStrength [38], diseñado para el análisis de opiniones, para detectar tendencias políticas, consiguiendo superar el rendimiento de un clasificador al azar. Ello sirve de punto de partida para incorporar información y datos que permitan detectar el sentimiento de los discursos expresados por partidarios de las principales ideologías políticas en España. En sus conclusiones destacan las diferencias existentes entre estas tendencias en función del *status quo* y el clima político. Por su parte Pla y Hurtado [32] plantean la siguiente hipótesis: opiniones favorables a un partido político por parte de un usuario indican una afinidad del autor a la ideología de dicho partido, mientras que opiniones en contra probablemente representen un desacuerdo. Para abordar el reto,

parten primero de los tuits de un usuario e identifican las entidades[2], mediante herramientas proporcionadas por FreeLing [2] o elementos de Twitter que tienen una alta probabilidad de representar a una entidad, como son los nombres de usuarios o hashtags. Del conjunto de entidades extraídas los autores seleccionan aquellas que contienen las siglas de alguno de los principales partidos y personalidades políticas del momento: CIU, IU, PP, PSOE, UPYD, Cayo Lara, Rajoy, Rubalcaba, Zapatero y González Pons. A continuación asignan un valor de tendencia política para las entidades de cada ideología: -1 a las de izquierda, +1 a las de derecha y 0 a las de centro. Así, los valores de polaridad y tendencia política para los tuits de un usuario son combinados para determinar su tendencia política.

[2]Segmento de texto que hace referencia a un nombre de persona, organización, localizaciones, expresiones, etc.

Capítulo 3

Recursos y herramientas utilizadas

En este capítulo se describen brevemente las principales tecnologías empleadas durante el desarrollo de este proyecto fin de máster.

3.1. Bibliotecas

Las bibliotecas descritas bajo estas líneas proporcionan principalmente funcionalidades de procesamiento del lenguaje natural de textos escritos y de acceso a los mensajes web necesarios para implementar y evaluar nuestra propuesta.

3.1.1. MaltParser

MaltParser[1] [28] es un generador de analizadores sintácticos de dependencias, implementado en Java y desarrollado por la Universidad de Växjö y la Universidad de Uppsala. A partir de un conjunto de entrenamiento, donde las oraciones estén etiquetadas como grafos de dependencias, se entrena un modelo de forma supervisada que permita analizar posteriormente nuevas frases. Se caracteriza por implementar distintos algoritmos de análisis, pudiendo configurarse para escoger cualquiera de ellos.

Los modelos generados por MaltParser presentan un buen rendimiento en una variedad de lenguajes como el inglés, francés, sueco o castellano. Es, sin embargo, necesario configurar y modificar distintos parámetros si se quiere optimizar la precisión para un idioma en concreto.

[1] http://www.maltparser.org/ Visto por última vez en agosto de 2012.

3.1.2. Natural Language Toolkit

Natural Language Toolkit (NLTK) es una plataforma de código abierto pensada para desarrollar programas Python relacionados con el PLN. NLTK nació con fines didácticos, pero ha sido adoptado muy rápidamente tanto en la industria como en el ámbito de la investigación, tanto por su eficiencia como por su sencillez de utilización y aprendizaje.

3.1.3. WEKA

WEKA es una colección de algoritmos de aprendizaje automático. Desarrollado por la Universidad de Waikato, esta biblioteca puede ser utilizada tanto mediante un entorno gráfico como desde líneas de comandos, o importando la propia biblioteca desde código Java. Esta biblioteca es habitualmente utilizada como apoyo en tareas de minería de datos. Entre sus principales funcionalidades se encuentran: la selección de características, mecanismos de preprocesamiento, así como algoritmos y técnicas de clasificación, regresión y clustering.

3.1.4. API REST de Twitter

Twitter proporciona una API REST para descargar tuits que cumplen unas determinadas condiciones u obtener información y estadísticas sobre los usuarios. Esta API es utilizada para la construcción de corpus propios para resolver los problemas de clasificación propuestos en este trabajo fin de máster.

3.2. Recursos lingüísticos

El desarrollo de este trabajo fin de máster se apoya en una serie de recursos lingüísticos descritos a continuación.

3.2.1. Corpus TASS

El corpus TASS es una colección de tuits escritos en castellano por diversas personalidades públicas. Dicha colección fue presentada en el Taller de Análisis del Sentimiento celebrado en la SEPLN[2][44]. Cada tuit se encuentra asignado a una o más las siguientes categorías (dado que un mismo mensaje puede abordar varios temas): *política, entretenimiento, economía, música, fútbol, tecnología, deportes* (distintos de fútbol), *literatura* y una categoría adicional *otros*, para representar los textos que pertenecen a otros temas. Es posible asignar más de

[2]Sociedad Española para el Procesamiento del Lenguaje Natural

una categoría a cada tuit porque no es extraño que los usuarios relacionen varias temáticas en un mismo mensaje.

Por otro lado, para cada uno de los usuarios presentes en este corpus se ha asignado un tendencia política, en función del contenido de sus mensajes. Así, es posible distinguir hasta cuatro ideologías políticas distintas: *derecha*, *centro*, *izquierda* o *indefinida* (en el caso de que el autor no manifieste su tendencia política en sus mensajes).

3.2.2. Corpus Ancora

Ancora es una colección bilingüe de documentos de más de 500 000 palabras disponible tanto para el castellano como para el catalán, desarrollada por el grupo de investigación CliC[3]. Se encuentra disponible vía web[4], previo registro. Cada una de las oraciones del texto se encuentra anotada como un árbol de dependencias en formato CoNLL-X [8]. Este formato de representación describe la estructura sintáctica de las oraciones en forma de árboles sintácticos de dependencias. Cada palabra contiene además información morfológica para cada uno de los términos, incluyendo categoría gramatical, género o número.

3.2.3. Diccionarios del LIWC

Linguistic Inquiry and Word Count (LIWC) [30] constituye un software de análisis lingüístico y automático de textos, que incluye una colección de diccionarios, disponibles también para el castellano. Estos diccionarios relacionan términos con *propiedades psicométricas* que reflejan distintas dimensiones del lenguaje humano. Así, se asocian términos con categorías psicológicas que estos transmiten como *ansiedad*, *ira* o *emociones positivas*. También se recogen otras propiedades que representan estos términos, incluyendo los temas a los que hacen referencia (*fútbol*, *economía* o *dinero*) e incluso información gramatical del propio término, indicando se se trata de un elemento que hace referencia o un evento pasado, presente o futuro.

[3]Centre de Llenguatge i Computació
[4]http://clic.ub.edu/corpus/

Capítulo 4

Procesamiento del lenguaje natural para el análisis de textos web

En este capítulo se presentan las etapas de procesamiento de lenguaje natural aplicadas como paso previo al análisis de textos web. Primero se llevan a cabo las fases de preprocesado, segmentación y tokenización, para luego llevar a cabo un análisis morfosintáctico de los textos que permita obtener la estructura sintáctica de sus oraciones.

4.1. Preprocesamiento

El preprocesador compila una colección de expresiones regulares creadas para tratar con algunas características del lenguaje, así como algunos de los elementos no gramaticales más habituales en entornos web:

- *Unificación de expresiones compuestas habituales.* En castellano, existen grupos de palabras que suelen actuar como una sola unidad de significado. Ejemplo de ello son expresiones como *'mientras que'* o *'al menos'*. Partiendo de un diccionario que recoja este tipo de expresiones, es posible recopilar una colección de expresiones regulares que las detecten y unifiquen en un solo término. Así, la expresión *'mientras que'* es convertida en *'mientras_que'*.

- *Normalización de los signos de puntuación.* En entornos web es habitual que los usuarios omitan los signos de puntuación como el '.'o ','. En otras ocasiones, aún no omitiéndolos, son utilizados de manera incorrecta (*e.g. 'No estoy de acuerdo,pero lo acepto.Lo pensaré'*) por diversas razones, como puede ser la falta la falta de espacio, en el ca-

so de Twitter. Mediante expresiones regulares, el preprocesador localiza los signos de puntuación mal colocados, separando mediante espacios en blanco cuando sea necesario (*e.g. 'No estoy de acuerdo, pero lo acepto. Lo pensaré'*). Situaciones especiales donde los signos de puntuación tienen objetivos distintos al habitual (como la representación de números decimales, por ejemplo), son detectadas para ser descartadas.

- *Normalización de abreviaturas más habituales*: Como en el caso de los signos de puntuación, dada la falta de espacio de la que disponen los usuarios en entornos web para expresar sus opiniones, es habitual encontrarse con abreviaturas no reconocidas. Dado un diccionario, con los términos reconocidos y un conjunto de textos web, es posible extraer un listado de palabras con los términos no presentes en el diccionario. A continuación, mediante una revisión manual de dichos términos, es posible separar las palabras desconocidas de formas abreviadas de palabras reconocidas, proporcionando una alternativa gramatical para estas últimas, que son recopiladas en un léxico de transformaciones. Así elementos como *'x'* o *'q'* son reemplazados por *'por'* o *'que'*, respectivamente.

- *Normalización de interjecciones*: La forma en la que los usuarios expresan risas en Internet es muy irregular (*e.g. 'jajjjjaja', 'JIJIJII'*), lo que puede resultar en problemas de dispersión si se pretende usar este elemento como una característica a la hora de clasificar textos. El preprocesador normaliza estas representaciones al patrón *jxjx*, donde $x \in \{a, e, i, o, u\}$.

- *Sustitución de emoticonos*: Como en el caso de las interjecciones, en muchas ocasiones los usuarios reflejan sus estados de ánimo en la red mediante multitud de emoticonos con distintas connotaciones (*':)', ':-o', '¬¬'*). Se utiliza la lista proporcionada en [1] para identificar los emoticonos más frecuentes, para dividirlos en cinco grupos: *muy positivos* (EMP), *positivos* (EP), *neutros* (ENEU), *negativos* (EN) y *muy negativos* (EMN). Cuando un emoticono es detectado este es reemplazado por el nombre de su grupo. El resultado constituye una oración independiente del resto del mensaje.

En el ejemplo 4.1.1 se ilustra el comportamiento del preprocesador para algunos ejemplos didácticos.

Ejemplo 4.1.1. El preprocesador genérico convertiría la oración *'Me voy.De vacaciones.A partir de la semana que viene jajjja :-D.'* a su versión normalizada *'Me voy. De vacaciones. A partir de la semana que viene jaja. EP.'*.

■

Además se consideran aspectos particulares de la jerga de Twitter:

- *Nombres de usuarios ('@')*. En esta red social, cada usuario dispone de un alias, precedido del símbolo *'@'* (*e.g. '@usuarioinventado'*). Sin embargo, símbolos como este pueden suponer problemas en términos de segmentación de palabras o análisis morfológico, dado que es un elemento no gramatical, característico de este medio. La estrategia del algoritmo de preprocesamiento para tratar con este fenómeno se basa en transformar los nombres de usuario a verdaderos nombres propios; eliminando el símbolo *'@'* y convirtiendo en mayúscula la primera letra. Así, los nombres de usuarios son convertidos a nombres propios desde un punto de vista gramatical.

- *Eliminación de hashtags ('#')*. Los hastags son términos incluidos en Twitter que los usuarios preceden del símbolo *'#'*, con el objetivo de etiquetar sus mensajes. Al hacer click sobre un hashtag el usuario es redireccionado al conjunto de tuits que contienen la misma etiqueta. Sin embargo, es habitual que el período de vida de los hashtags sea muy corto, dado que suelen referir eventos muy específicos (*e.g. '#Goya2014', '#SuperBowlWinner'*). Por ello, este tipo de hashtags que sirven para clasificar según eventos concretos, y que son situados bien al principio o al final del tuit, son eliminados. También es frecuente utilizarlos como medio para enfatizar una palabra contenida en un mensaje (*e.g 'La #felicidad es algo difícil de conseguir'*). En este caso, únicamente el símbolo *'#'* es borrado.

- *Simplificación de enlaces*: En esta red social es habitual que los usuarios enlacen recursos externos, como imágenes o direcciones a otras páginas web. Con el fin de normalizar todas estas URL, se utilizan expresiones regulares para detectarlas y sustituirlas por el texto *'url'*. Con ello no se persigue una normalización gramatical sino, tratar de normalizar estos elementos que pueden servir de ayuda de entrada al clasificador, como veremos en capítulos siguientes.

El ejemplo 4.1.2 muestra algunos ejemplos para mensajes hipotéticos en este red social.

Ejemplo 4.1.2. Dada una oración *'La importancia de tener #ideas de #negocio propias como @usuario #emprendedor'* se obtendría como resultado: *'La importancia de tener ideas de negocio propias como Usuario'* tras aplicar nuestro algoritmo de preprocesamiento.

■

Como se comentó previamente, el preprocesado pretende corregir, de manera *ad-hoc*, algunos de los problemas habituales en entornos de web, de cara a incrementar el rendimiento de las tareas abordadas en este trabajo fin de máster. Aspectos más complejos como el tratamiento de tildes, o la inclusión, eliminación o transposición de caracteres en un término para su corrección, no son contemplados.

La figura 4.1 ilustra el diseño UML de alto nivel para la jerarquía del preprocesador, implementada siguiendo un patrón Decorador.

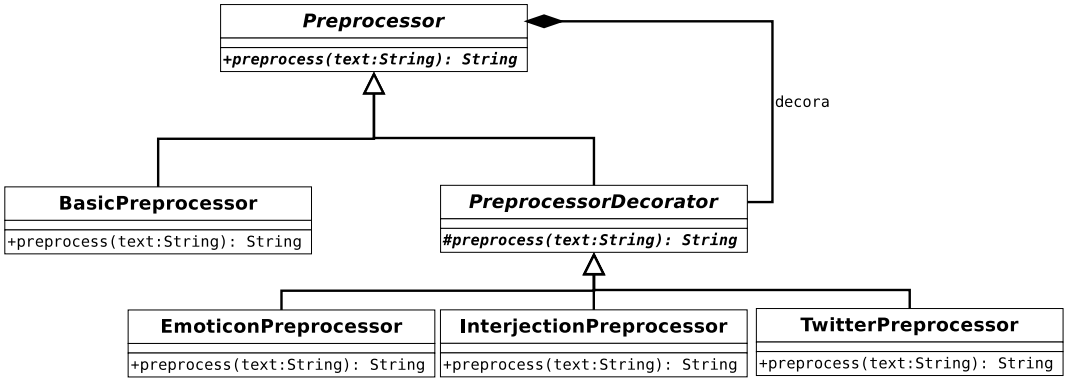

Figura 4.1: Jerarquía de clases de *Preprocessor*

4.2. Segmentación de frases y palabras

Tras el preprocesado del texto ya se han normalizado alguno de los elementos que podían complicar la identificación y división de frases y palabras, como la incorrecta colocación de signos de puntuación o la aparición de nombres de usuarios de Twitter o hashtags. Para la segmentación de oraciones este trabajo delega en la biblioteca NLTK y concretamente en los segmentadores de oraciones[1] que vienen entrenados en su directorio *nltk_data*. Estos modelos se caracterizan por utilizar un algoritmo de entrenamiento no supervisado que construye un modelo en el que se tienen en cuenta abreviaturas, siglas, colocaciones y otras palabras que representan el inicio de una oración. Con ese modelo, se intenta separar las distintas oraciones de un texto buscando delimitadores.

Ejemplo 4.2.1. El tuit *'La quinta ensaladera, Nadal saludando al equipo argentino. Y ahora al público. Oe, oe, oe.'*, es un ejemplo real de un mensaje en esta red social. Este ejemplo servirá como punto de partida para explicar otros aspectos del sistema en la medida de los posible. Tras la segmentación de las oraciones obtendríamos:

[1]Instancias serializadas de la clase *nltk.tokenize.punkt.PunktSentencetokenizer*

$$O_1 = \textit{‘La quinta ensaladera, Nadal saludando al equipo argentino.’} \tag{4.1}$$

$$O_2 = \textit{‘Y ahora al público.’} \tag{4.2}$$

$$O_3 = \textit{‘Oe, oe, oe.’} \tag{4.3}$$

■

Tras este paso, se lleva a cabo la segmentación de palabras. Existen diversos enfoques para realizar esta tarea. Algunos de ellos delegan en estrategias triviales como la separación por espacios en blanco. Esto, no obstante no es un mecanismo adecuado para segmentar los términos de lenguajes natural. Por ejemplo, con esta estrategia, la segmentación de la frase *‘Me gustó la película, pero es demasiado larga’* daría como resultado una lista de 8 términos donde el cuarto sería *‘película,’*, lo cual no sería correcto. En el ejemplo se observa que uno de los principales problemas de la segmentación de palabras es determinar la estrategia a seguir con símbolos como la coma. La biblioteca NLTK incluye segmentadores capaces de tratar con lenguajes naturales considerando sus peculiaridades mediante una colección de expresiones regulares, como el *nltk.tokenize.PunktWordTokenizer* (utilizado para este proyecto fin de máster).

Ejemplo 4.2.2. Para la oración 4.1 del ejemplo 4.2.1 la salida sería:

[*‘La’, ‘quinta’, ‘ensaladera’, ‘,’, ‘Nadal’, ‘saludando’, ‘al’, ‘equipo’, ‘argentino’, ‘.’*]

■

4.3. Análisis morfológico

Dada una oración $S = w_1 w_2 ... w_{n-1} w_n$, donde cada w_i se corresponde con un término, y conjunto de etiquetas morfológicas $T = \{t_1, t_2 ... t_{m-1} t_m\}$ el proceso de análisis morfológico, también conocido como *etiquetación*, consiste en asignar a cada término su correspondiente etiqueta, creando una lista de tuplas (s, t) con $s \in S$ y $t \in T$.

La etiquetación es una tarea compleja y propensa a errores, dado que una palabra puede tener una etiqueta distinta dependiendo del contexto. En el ejemplo 4.3.1 se ilustra un caso de esta situación.

Ejemplo 4.3.1. En castellano existen multitud de palabras ambiguas cuyo significado cambia en el contexto en el que se utilicen. La palabra *'juego'* es uno de esos ejemplos. Así, dicho término actúa como un *sustantivo* en:

'El fútbol es un juego'

pero no es difícil imaginar situaciones en las que el mismo término realice la función de un *verbo*:

'Juego al fútbol los miércoles'

■

Existen distintos distintos enfoques para llevar a cabo una etiquetación morfológica. La utilización de prefijos, sufijos y expresiones regulares ha sido una aproximación clásica a la hora de determinar la categoría gramatical de una palabra siguiendo un enfoque no supervisado. En la actualidad es muy frecuente emplear mecanismos de etiquetación estadísticos basados en n-gramas[2]. Markov fue el primero en proponer este tipo de técnica, utilizando bigramas y trigramas para tratar de determinar si la siguiente letra de una palabra sería vocal o consonante. Sus estudios permitieron desarrollar toda una rama de etiquetadores estadísticos capaces de calcular la secuencia de etiquetas más probable dentro de una oración [5, 25], lo que otorga cierta capacidad contextual al proceso. En los últimos años, un enfoque más popular consiste en interpretar la etiquetación como un proceso genérico de clasificación, donde a cada término dentro de su contexto se le asigna una de las distintas clases de etiquetas posibles. La palabra se clasifica en base a una serie de características de su entorno[3], permitiendo tener en cuenta tanto a los términos que la preceden como a los que la siguen. En documentos correctamente escritos el rendimiento de este tipo de métodos es realmente bueno, pero su precisión empeora de forma sensible cuando trabajan con textos donde abundan las palabras desconocidas. Un enfoque alternativo a los modelos de n-gramas lo constituye el basado en transformaciones, comúnmente conocida como etiquetación de Brill [6]. Esta propuesta se basa en el aprendizaje basado en transformaciones y dirigido por error para tratar de asignar la etiqueta correcta a cada palabra. En la implementación original del algoritmo, se distinguen dos fases. La primera es de inicialización. Aquí, a cada palabra se

[2]Un n-grama es como una secuencia de constituyentes gramaticales en una misma frase. Estos pueden ser palabras sílabas o simplemente caracteres.

[3]Como características tipográficas o las propias etiquetas

le asigna en un primer momento su etiqueta más probable sin considerar el contexto en el que aparece. En caso de que se trate de una palabra desconocida, se tratará como un nombre propio si comienza por una letra mayúscula y si no se etiquetará como un sustantivo común. A continuación se lleva a cabo la fase de aprendizaje. Partiendo de una serie de reglas contextuales y reglas heurísticas de etiquetación de palabras desconocidas, se lleva a cabo un proceso iterativo en el que se seleccionan las reglas candidatas que mejoren la precisión del etiquetador y que se resume en los siguientes pasos:

1. Medir el número de errores antes y después de aplicar de forma separada cada una de las reglas.

2. Seleccionar la regla que mejore más el rendimiento.

3. Añadir la regla candidata al conjunto de reglas, y etiquetar de nuevo el texto.

4. Repetir el proceso hasta que no haya ninguna regla que mejore la precisión actual del etiquetador.

Esta aproximación ha sido en la que se ha basado este trabajo dado su buen rendimiento en entornos en los que muchas palabras son desconocidas, como es el caso de la web, donde las normas ortográficas son frecuentemente ignoradas y el uso de vocablos agramaticales o no disponibles en el diccionario es habitual. En concreto, para entrenar nuestro modelo se delegó en la implementación del etiquetador de Brill [6] incluida en el NLTK [31, 37]. La implementación proporcionada en esta biblioteca permite que Brill reciba una etiquetador base, a partir del cual aplicar las reglas de corrección. Una de las diferencias de esta implementación respecto a la original, es la capacidad de asignar un etiquetador base que realice una primera anotación, en lugar de asignar simplemente la etiqueta más probable. Como etiquetador base se optó por un modelo de procesamiento estadístico basado en tri-gramas. Ancora fue el corpus usado para entrenar nuestro analizador morfológico. Se utilizó el 90 % de la colección para entrenar los modelos y el restante 10 % para medir su rendimiento. El rendimiento teórico obtenido fue de un 95,86 % lo que es coherente con el estado del arte para el castellano. Sin embargo, se observó empíricamente que su rendimiento cuando el modelo etiquetaba textos web era pobre. Para solucionarlo, se optó por clonar el corpus Ancora. Cada texto de la colección fue replicado para generar un texto equivalente, donde todas las tildes son omitidas. Este corpus ampliado fue el utilizado para entrenar la versión definitiva de nuestro etiquetador. La tabla 4.1 ilustra un ejemplo real de salida de los etiquetadores entrenados sobre el corpus

estándar y sobre el duplicado para la oración: *'No he tenido tiempo de escribir sobre el y ya esta estropeado'*, donde el pronombre *'el'* y el verbo *'esta'* presentan errores ortográficos. Se observa que el etiquetador estándar falla esos dos términos, mientras que el clasificador expandido si los anota correctamente.

	No	he	tenido	tiempo	de	escribir	sobre	el	y	ya	esta	estropeado
M	r	v	v	n	s	v	s	p	c	s	v	v
E	r	v	v	n	s	v	s	**d**	c	s	**d**	v

Tabla 4.1: Ejemplo de etiquetación morfológica para el etiquetador modificado (M) y el entrenado con la versión estándar de Ancora (E): adverbio (r), verbo (v), nombre (n), preposición (s), pronombre (p), conjunción (c), determinante (d)

4.4. Análisis sintáctico de dependencias

El análisis sintáctico de dependencias es un proceso que permite extraer las relaciones entre los componentes de una oración, lo que contribuye a comprender e interpretar eficazmente un texto, completando el análisis léxico precedente y sirviendo de partida para el semántico posterior. Sea $S = [p_1 p_2 ... p_{n-1} pn]$ una oración donde p_i representa a cada una de las palabras de la oración, el resultado de aplicar un analizador sintáctico de dependencias es un grafo G, constituido por *tripletas de dependencia* de la forma (p_i, arc_{ij}, p_j), donde p_i representa el término *padre*, p_j el *dependiente*, y arc_{ij} la función sintáctica que relaciona a ambas palabras, denominada *tipo de dependencia*. Por cuestiones formales y de implementación es habitual incluir un primer término artificial, w_0, que actúa siempre como la raíz del grafo, denominado ROOT. La figura 4.2 ilustra una salida válida para un análisis sintáctico de dependencias para la primera oración de nuestro ejemplo: *'La quinta ensaladera, Nadal saludando al equipo argentino.'*.

Para entrenar nuestro analizador sintáctico de dependencias se ha empleado el software MaltParser [28], descrito en capítulos anteriores. MaltParser soporta el formato de representación estándar para los ficheros de entrada, denominado CoNLL-X, y que permitirá entrenar el modelo para luego evaluar nuevas entradas. Este formato se ilustra en la tabla 4.2 para nuestro ejemplo, donde cada columna representa:

1. ID: La posición dentro de la oración. Se reserva la posición 0 para el *token* extra corres-

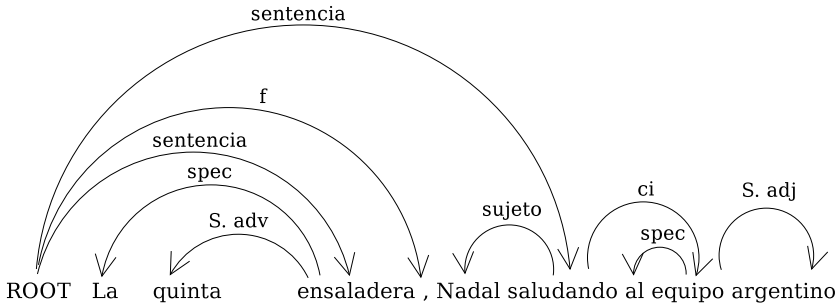

Figura 4.2: Ejemplo de un análisis sintáctico de dependencias

pondiente al ROOT.

2. FORM: El propio *token*.

3. LEMMA: La forma canónica de la palabra. Por ejemplo, para la palabra *"saludando"*, el lema sería *saludar*.

4. CPOSTAG: Etiqueta de grano grueso con información léxica muy general, comúnmente su categoría gramatical. Así, *Verbo* sería la etiqueta de grano grueso de *"saludando"*.

5. POSTAG: Etiqueta de grano fino. Suele añadirse información léxica que complementa a CPOSTAG. *Verbo_gerundio* podría ser una etiqueta de grano fino para el ejemplo anterior.

6. FEATS: Conjunto de información sintáctica y morfológica. Por ejemplo, para el caso de *"saludando"*, unas FEATS válidas consistiría en indicar que se trata de un verbo de la primera conjugación.

7. HEAD: Indica el ID del *token* del que depende, esto es, su padre.

8. DEPREL: Indica el tipo de dependencia que se mantiene con HEAD.

El rendimiento obtenido por el analizador sintáctico alcanzó un LAS[4] de 83.75 %, un UAS[5] de 88.16 % y un LA[6] de 88.61 %, lo cuál es coherente con el estado el arte para el castellano.

[4] LAS (*Labeled Attachment Score*): Métrica que mide el porcentaje de palabras a las que tanto el padre como el tipo de dependencia fueron asignados correctamente

[5] UAS(*Unlabeled Attachmente Score*): Métrica que sólo tiene en cuenta que el padre de la relación de dependencia esté bien asignado

[6] LA(*Label Accuracy Score*): Métrica que mide el porcentaje que mide los tipos de dependencia asignados correctamente

ID	FORM	LEMMA	CPOSTAG	POSTAG	FEATS	HEAD	DEPREL
1	La	el	d	d_articulo	masc_sing	3	spec
2	quinta	quinto	r	orden	_	3	S. adv
3	ensaladera	ensaladera	n	n_común	fem_sing	0	sentencia
4	,	,	f	f	_	0	f
5	Nadal	Nadal	n	n_propio	_	6	sujeto
6	saludando	saludar	v	v_gerundio	1ª conj	0	sentencia
7	al	al	s	s_contraída	_	8	spec
8	equipo	equipo	n	n_comun	masc_sing	6	ci
9	argentino	argentino	a	a_calificativo	masc_sing	8	S. adj

Tabla 4.2: Ejemplo de un grafo de dependencias en formato CoNLL-X

4.5. Diagrama de flujo de procesamiento del lenguaje natural utilizado

La figura 4.3 ilustra gráficamente el diagrama de flujo seguido, una vez se obtiene el contenido original de un tuit. En los próximos capítulos describiremos como emplear los árboles de dependencias obtenidos para abordar las tareas de clasificación de tópicos e identificación de tendencias políticas de usuarios.

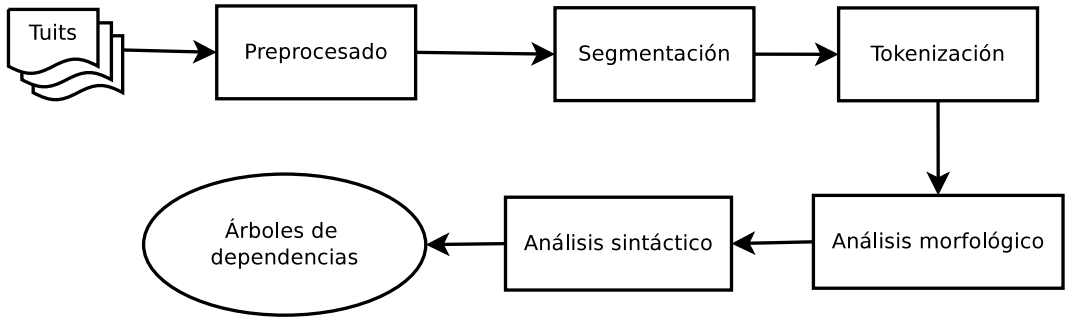

Figura 4.3: Arquitectura general del sistema de procesamiento de lenguaje natural

Capítulo 5

Clasificación de tópicos en Twitter

Este capítulo describe el enfoque propuesto en este trabajo para identificar las temáticas que trata un tuit. En primer lugar, se presenta la naturaleza del problema desde el punto de vista de clasificación. A continuación se describen los modelos iniciales considerados para resolver el problema, incluyendo tanto aproximaciones léxicas como sintácticas.

5.1. Enfoque multi-etiqueta

Son varias las situaciones que se pueden dar al tratar de resolver un problema de clasificación. La situación más simple es aquella consistente en una *clasificación binaria*, donde se decide si una instancia pertenece a una clase o no. Aunque útil para resolver ciertos problemas, este enfoque es a menudo insuficiente. No es extraño que una aplicación deba distinguir entre varias clases con diferentes características, lo que comúnmente se denomina *clasificación multi-clase*. Dada una colección de n categorías, el modelo entrenado debe determinar a cuál de ellas pertenecen los elementos de la muestra. No se debe confundir un problema de clasificación multi-clase con un problema de clasificación *multi-etiqueta*. En el primer caso, una instancia únicamente puede ser asignada a una etiqueta, mientras que en el segundo es posible asignar un elemento de la muestra a más de una categoría.

La clasificación de tópicos debe ser abordada como una tarea de clasificación multi-etiqueta. No es extraño que los usuarios relacionen distintos temas cuando expresan sus pareceres en mensajes, como se observa en los siguientes ejemplos de tuits reales:

- *'La clave del nuevo gobierno, su estructura ¿Habrá dos vicepresidencias o ninguna? La clave, en el equipo económico'*. Leyendo este tuit parece razonable concluir que son dos los tópicos que se relacionan: *política* y *economía*. Se trata de un caso sencillo

donde términos como *'gobierno'*, *'vicepresidencia'* o *'económico'* permiten identificar sin complicaciones el argumento del mensaje.

- *'El público inteligente está en las redes sociales. La educación determina su uso más que la riqueza. Impacto en medios'*. Se trata de un ejemplo aún más completo donde podríamos determinar que se están relacionando hasta tres temáticas distintas: *tecnología* y *entretenimiento*, al referir la expresión *'redes sociales'*, muy asociada a la Web 2.0 y a la manera en que interactúa y se comunica la sociedad en la actualidad; y *economía* por la utilización del término *'riqueza'*. No sería extraño tampoco concluir que este tuit podría asignarse a más categorías como *educación*. En este ejemplo podemos observar por tanto dos problemas de la tarea que nos ocupa. El primero es referido a la necesidad de considerar términos compuestos en lugar de palabras sueltas para poder extraer conclusiones sobre lo que se está hablando. El segundo trata sobre la complejidad, no solo para reconocer ciertos tópicos, sino de la dificultad para establecer un consenso para determinar que tópicos deberían identificarse.

- *'Holaaa mis tweeps! Un día maravilloso para hacer un twitcam como os he prometido sera a las 6:00pm (hora de Mexico) nos vemos pronto'*. Este es un ejemplo que ahonda en alguna de las dificultades derivadas de esta tarea. Un experto en anotación de corpus asignó este tuit dentro de la categoría *música*, cuando en apariencia este mensaje no parece tratar nada en concreto. Sin embargo, se trata de un tuit escrito por el cantante Alenjandro Sanz, y los expertos consideraron que por esa razón dicho mensaje debería encontrarse en esa categoría. Se trata de un criterio discutible, ya que siguiendo ese patrón todos los tuits de un cantante deberían ser catalogados como musicales. Y por la misma razón todos los tuits de un político deberían ser catalogados como políticos, algo que de nuevo no siempre es cierto. En este trabajo no se sigue este último criterio y los mensajes son analizados independientemente de la información que se disponga sobre los usuarios.

Existen distintos enfoques a la hora de abordar una tarea de clasificación multi-etiqueta. Sea un problema de clasificación que desea asignar instancias a uno o mas de los elementos de un un conjunto N de categorías; son dos las principales estrategias utilizadas a la hora de resolver este problema:

- *Conversión a un problema multi-clase*. Dado el conjunto N, este enfoque propone crear

un nuevo conjunto de clases, N' resultante de crear combinaciones sin repetición de 1 a |N| elementos. El ejemplo 5.1.1 ilustra un sencillo ejemplo de este enfoque.

- *Estrategia uno contra todos.* Esta perspectiva propone entrenar |N| clasificadores, donde cada uno de ellos permite diferenciar un tópico i, donde $i \in N$, de todos los demás. El ejemplo 5.1.2 describe un ejemplo sobre como construir un modelo basado en esta estrategia.

Ejemplo 5.1.1. Un problema de clasificación multi-etiqueta donde $N = \{X, Y, Z\}$, sería transformado en un problema de clasificación multiclase donde tras obtener las combinaciones sin repetición de los elementos del conjunto N, obtendríamos un conjunto $N' = \{X, Y, Z, XY, XZ, YZ, XYZ\}$. Así, pasaríamos a tener un problema de clasificación multiclase que podría abordarse como hemos explicado previamente.

■

Ejemplo 5.1.2. Continuando con el ejemplo anterior con las categorías X, Y y Z, sería necesario construir tres clasificadores para resolver la el problema de clasificación mediante una estrategia uno contra todos: X *vs* N-$\{X\}$, Y *vs* N-$\{Y\}$ y Z *vs* N-$\{Z\}$.

■

Ambas estrategias presentan ventajas e inconvenientes. Respecto a la transformación de multi-etiqueta a multi-clase destaca el crecimiento en el número de clases en N' cuando N es grande. Ello puede repercutir en que la muestra que se utilice para entrenar el clasificador supervisado disponga de pocos ejemplos para algunas de las combinaciones resultantes en N', y el modelo entrenado no aprenda a diferenciar esas categorías, centrándose únicamente en las mayoritarias. En relación a la estrategia uno contra todos, el principal inconveniente reside en la necesidad de crear y ejecutar |N| clasificadores para obtener a que categorías pertenece un ejemplo de la muestra. Ello resulta en un mayor consumo de recursos y tiempo de ejecución. En este trabajo fin de máster se ha optado por la segunda estrategia, dado que el corpus entrenado para construir los clasificadores distingue hasta 10 tópicos, creciendo el número de combinaciones considerablemente. Además, como se indica en el Apéndice A, para muchas categorías combinadas el número de muestras es muy pequeño, complicando que el clasificador pueda aprender esas clases.

5.2. Modelos base

Tomando el documento como unidad de análisis, tuits en el caso de este trabajo, los modelos base descritos a continuación tienen en cuenta distintos conjuntos de atributos o características, contando el número de apariciones que se da en el texto para cada una de ellas. Estas características permitirán entrenar clasificadores supervisados que detecten el total de las temáticas a las que hace referencia el tuit.

5.2.1. Modelos de ngramas

La utilización de n-gramas de términos es uno de los enfoques más habituales a la hora de resolver muchas tareas de análisis automático de textos. El modelo más básico que es posible construir siguiendo este enfoque es el denominado *bolsa de palabras*. Dado un documento, éste es segmentado obteniendo como resultado una colección de términos, donde cada uno de ellos actúa como un atributo de entrada al clasificador. Este modelo suele constituir, en muchas ocasiones, una buena línea de base que obtiene un rendimiento aceptable. Es común aplicar sencillas técnicas de normalización como la conversión a minúsculas para disminuir el espacio dimensional de características y agrupar aquellas que representaran lo mismo. El ejemplo 5.2.1 ilustra cómo funciona una bolsa de palabras básica para nuestro ejemplo.

Ejemplo 5.2.1. Dado el texto *'La quinta ensaladera, Nadal saludando al equipo argentino. Y ahora al público. Oe, oe, oe.'* el resultado de nuestra bolsa de palabras sería {*'La', 'quinta', 'ensaladera', ',', 'Nadal', 'saludando', 'al', 'equipo', 'argentino', '.', 'Y', 'ahora', 'al', 'público', 'oe'*}, donde *'oe'* y *','* tienen un valor de ocurrencia de 3, el *'.'* de 2 y el resto de términos de 1.

■

Una manera para tratar de reducir derivados de estas ambigüedades consiste en emplear n-gramas de mayor longitud, como bi-gramas, secuencias de dos palabras consecutivas. Es una manera de capturar cierta información estructural sobre cómo las palabras se relacionan sin necesidad de aplicar técnicas de análisis sintáctico. Las relaciones entre términos son en su mayoría cercanas, lo que habitualmente permite aplicar este tipo de optimizaciones de manera satisfactoria. Uno de los principales problemas de la utilización de n-gramas de mayor longitud es la dispersión. El espacio dimensional de las características de entrada, cuando el tamaño de los n-gramas crece, explota exponencialmente. Ello provoca que muchos de los atributos sólo aparezcan unas pocas veces en el conjunto de entrenamiento y el clasificador

no pueden aprender correctamente, disminuyendo el rendimiento. El ejemplo 5.2.2 explica este fenómeno.

Ejemplo 5.2.2. Dado el texto *'La quinta ensaladera, Nadal saludando al equipo argentino. Y ahora al público. Oe, oe, oe.'*, el resultado de nuestra bolsa de de bi-gramas sería[1] { *'La quinta', 'quinta ensaladera', ... 'oe oe',* }. Algunos de los bigramas de palabras que podemos obtener pueden ser muy informativos. Por ejemplo, *'quinta ensaladera'* o *'equipo argentino'*, son dos buenos indicadores de que un texto trate sobre *deportes*. No obstante, también será menos habitual encontrar estas cadenas de caracteres y no es difícil imaginar situaciones que probablemente representen lo mismo, pero con pequeñas variaciones como: *'equipos argentinos'*, *'equipación argentina'* o *'equipo español'*.

■

El castellano es un lenguaje con una fuerte riqueza morfológica, lo que se traduce flexiones de género y número para nombres, adjetivos o determinantes, e incluso de tiempo en el caso de los verbos. Ello puede derivar problemas de dispersión. Términos como *'juego'*, *'jugó'* o *'jugamos'* constituirían palabras distintas y por tanto características distintas de entrada al clasificador, algo que no parece tener sentido ya que probablemente esas tres palabras suelan emplearse para tratar sobre el mismo tema. El ejemplo 5.2.3 refleja como se trataría el ejemplo siguiendo esta sencilla técnica de normalización.

Ejemplo 5.2.3. Dado el texto *'La quinta ensaladera, Nadal saludando al equipo argentino. Y ahora al público. Oe, oe, oe.'* el resultado de nuestra bolsa de lemas sería *'El', 'quinto', 'ensaladera', ',', 'Nadal', 'saludar', 'al', 'equipo', 'argentino', '.', 'y', 'ahora', 'al', 'público', 'oe',* }, donde *'oe'* y *','* tienen un valor de ocurrencia de 3, el *'.'* de 2 y el resto de términos de 1.
Sin embargo, términos como *'saludando'* o *'quinta'* son reducidos a su forma canónica, obteniendo características más representativas para entrenar el clasificador.

■

5.2.2. Modelo morfológico

La utilización de la información morfológica de las palabras ha sido utilizada con éxito en otros ámbitos de la clasificación de textos web, como en la identificación de textos con y sin opinión. En [29] se indica que ciertas categorías gramaticales e información morfológica

[1] Se ilustran únicamente algunos de los elementos de la bolsa a modo didáctico

pueden ser buenos indicativos tanto de textos subjetivos como objetivos. Así, la presencia de adjetivos o adverbios releja la presencia de contenido con opinión o sentimiento, dado que son tipos de palabras utilizados para describir y calificar aspectos y conceptos. En el lado opuesto, utilizar verbos en tercera persona puede ser un indicio de textos informativos, ya que al expresar opiniones es habitual utilizar formas en primera o segunda persona, expresando un alto nivel de cercanía con lo que nos ha pasado. En lo referido a la identificación de tópicos, nuestra hipótesis es que no deberían ser muy útiles por sí mismas. Sin embargo, existen ciertas categorías gramaticales cuya frecuencia puede ser mayor en ciertos tipos de textos. Así, es probable que etiquetas como *nombre propio* sean más habituales en categorías como deportes o películas que en economía o agricultura. El ejemplo 5.2.4 ilustra como actuaría este modelo sobre nuestro texto de referencia.

Ejemplo 5.2.4. A partir de la oración *'La quinta ensaladera, Nadal saludando al equipo argentino. Y ahora al público. Oe. oe, oe.'* sería posible extraer (entre otras características) la siguiente información morfológica: 1 *nombre propio*, 2 *nombres comunes masculinos singular*, 1 *nombre común femenino singular*, 1 *verbo en gerundio* o 2 *adjetivos*. Como ya comentamos, parece difícil determinar correctamente la temática de un tuit a partir de esta información, en especial para un humano. Sin embargo muchas veces el clasificador es capaz de establecer conclusiones que pasan desapercibidas o non son consideradas por nuestra parte como indicativos de un tópico específico. Este aspecto será discutido más en detalle en el capítulo de resultados.

■

5.2.3. Modelo psicométrico

Este modelo toma como base de conocimiento los diccionarios de los que dispone el LIWC para el castellano. La tabla 5.1 presenta un pequeño fragmento de dicha colección. El objetivo es entrenar un modelo que aprenda a distinguir entre distintas categorías en función de los tópicos recogidos en este recurso como: *deportes*, *familia*, *dinero*, *trabajo* o *religión*. También se pretende medir cómo la influencia de ciertas emociones o aspectos psicológicos como la *ira*, la *tristeza* o la *alegría* pueden influir en la identificación de temáticas en textos web. A este respecto, nuestra hipótesis se basa en que ámbitos como la política son más proclives a mostrar emociones negativas en el clima actual de la sociedad española.

Uno de las debilidades de este modelo es que al depender totalmente de un recurso externo, sus capacidades están limitadas a la cobertura de los diccionarios en cuestión. Este

Término	Propiedades psicométricas
atletismo	*Deportes, Placer*
anuncio	*TV, Placer, Trabajo, Ocupación*
esposo	*Familia, Sentir*

Tabla 5.1: Fragmento de los diccionarios del LIWC para el castellano

ha sido un problema ya recogido por otros autores en otros ámbitos [47], y los experimentos recogidos en este trabajo confirman esta debilidad.

Ejemplo 5.2.5. Continuando con nuestro ejemplo didáctico *'La quinta ensaladera, Nadal saludando al equipo argentino. Y ahora al público. Oe. oe, oe.'* El modelo psicométrico sería capaz de extraer las siguientes propiedades psicométricas: *social*, *ocupación*, *trabajo*, *placer* y *deportes* de la palabra *'equipo'*, *tiempo* del término *'ahora'*; y de nuevo *social* de la palabra *'público'*.

■

5.2.4. Modelos de tripletas sintácticas generalizadas

La alternativa sintáctica a la utilización de los bigramas son las tripletas sintácticas, que representa una relación estructural entre dos términos, con la diferencia de que existe una función sintáctica que los conecta, en lugar de establecer esa relación únicamente en base a su proximidad. Este es uno de los puntos débiles de los bigramas, al asumir que los términos relacionados son siempre contiguos, algo que no tiene por que ser cierto. Por su parte, el punto fuerte del análisis de dependencias reside en su capacidad de relacionar elementos lejanos. El ejemplo 5.2.6 ilustra un caso que refleja las diferencias entre los bigramas estándar y los obtenidos mediante tripletas sintácticas.

Ejemplo 5.2.6. Dada la frase *'No me interesa'*, sus bigramas correspondientes serían: *'no me'* y *'me interesa'*. Si lo que se pretende es capturar la relación estructural entre términos, este resultado no sería correcto, dado que *'no'* debe estar relacionado con *'interesa'* y no con *'me'*. Sin embargo, un analizador sintáctico de dependencias si detectaría correctamente estaa relaciones obteniendo como resultado dos tripletas de la forma: $(interesa, neg, no)$ y $(interesa, ci, me)$ donde *neg* y *ci* representarían las funciones sintácticas de negación y complemento indirecto, respectivamente.

■

No obstante, las tripletas sintácticas, como los bigramas, sufren problemas de dispersión. Una posible solución a este fenómeno consiste en la utilización de tripletas sintácticas generalizadas. La propuesta inicial de este método fue presentada en [20]. Dada una tripleta de la forma (p_i, arc_{ij}, p_j), los autores proponen generalizar bien p_i o p_j (o bien los dos) a su categoría gramatical, con el objetivo de reducir la dispersión de las tripletas. Su propuesta fue utilizada para incrementar el rendimiento de sistemas encargados de identificar oraciones con opinión en críticas expresadas en la web de Amazon, obteniendo una mejora del rendimiento estadísticamente significativa, cuando las tripletas eran usadas conjuntamente con características léxicas. Nuestra hipótesis es que las tripletas sintácticas generalizadas también pueden ser de utilidad en la clasificación de tópicos. Sin embargo, creemos que la idea original de los autores puede ser enriquecida y mejorada. En primer lugar, el concepto inicial de tripletas sintácticas sólo contempla la generalización bien del padre o del dependiente a su correspondiente categoría gramatical. Ello pude suponer la pérdida de información relevante. Este trabajo propone una función de generalización enriquecida, *g(w,x)*, donde w indica el términos a generalizar y x el valor a devolver, incluyendo: el propio término, su forma lematizada, sus categorías psicométricas, o un token vacío, si se decide eliminar completamente el término en cuestión. También se contempla eliminar o no el tipo de dependencia. El ejemplo 5.2.7 ilustra el comportamiento de este tipo de tripletas, y como su utilización permite reducir la dispersión, al abstraer la representación de ciertos conceptos.

Ejemplo 5.2.7. Este ejemplo ilustra para algunas tripletas sintácticas básicas su correspondiente generalización, acorde con los criterios explicados en esta sección:

Generalización deseada		**Resultado**
(g(equipo, etiqueta), delete(modificador), argentino)	→	*(nombre común, _, argentino)*
(g(equipo, propiedades psicométricas), modificador, argentino)	→	*(ocio, modificador, argentino)*
		(deportes, modificador, argentino)
		...
(g(equipo, propiedades psicométricas), modificador, g(argentino, etiqueta))	→	*(ocio, modificador, adjectivo)*
		(deportes, modificador, adjectivo)
		...

Observando los ejemplos de la tabla, es aceptable suponer que tripletas de significado similar como *(fútbol, modificador ,argentino)* o *(jugador, modificador, argentino)* puedan ser generalizadas también como *(deportes, modificador, argentino)*, permitiendo unificar características de significado similar y reduciendo la dispersión para que el clasificador pueda crear modelos más efectivos.

■

5.3. Modelos combinados

Los modelos iniciales propuestos en este capítulo permiten conocer el rendimiento de conjuntos de características básicos, identificando sus puntos fuertes y débiles, como se indica en la sección de resultados experimentales. El objetivo de los enfoques combinados es el de crear modelos que sean entrenados a partir de conjuntos de características complejos, donde conocimiento léxico, psicométrico y semántico, es combinado, ya sea como características separadas, o relacionadas mediante información de carácter sintáctico, como se ilustra en el capítulo de resultados experimentales.

5.4. Arquitectura del selector de características

A nivel de implementación, la construcción de los modelos requiere la necesidad de detectar distintas propiedades en el texto que puedan ayudar al clasificador a identificar los tópicos, lo que incluye identificar desde palabras hasta pequeños subgrafos. El sistema aquí propuesto implementa una jerarquía *Counter*, encargada de detectar estos fenómenos. de alto nivel. La jerarquía de clases implementa una patrón composición con plantilla. El sistema soporta cuatro contadores básicos para analizar los textos: (1) *PsychometricCounter*, encargado de identificar las propiedades psicométricas, (2) *NGramCounter*, que detecta los ngramas de longitud deseada, (3) *PoStagCounter* que tiene la función de contar distintas propiedades morfológicas y (4) *DependencyTripletsCounter* un contador que identifica segmentos de grafos para utilizar su frecuencia como entrada posterior al clasificador. Por su parte, *CompositeCounter* permite agregar distintos contadores, para así disponer de la capacidad de crear posteriormente modelos combinados. La figura 5.1 ilustra el diseño UML para esta jerarquía de clases.

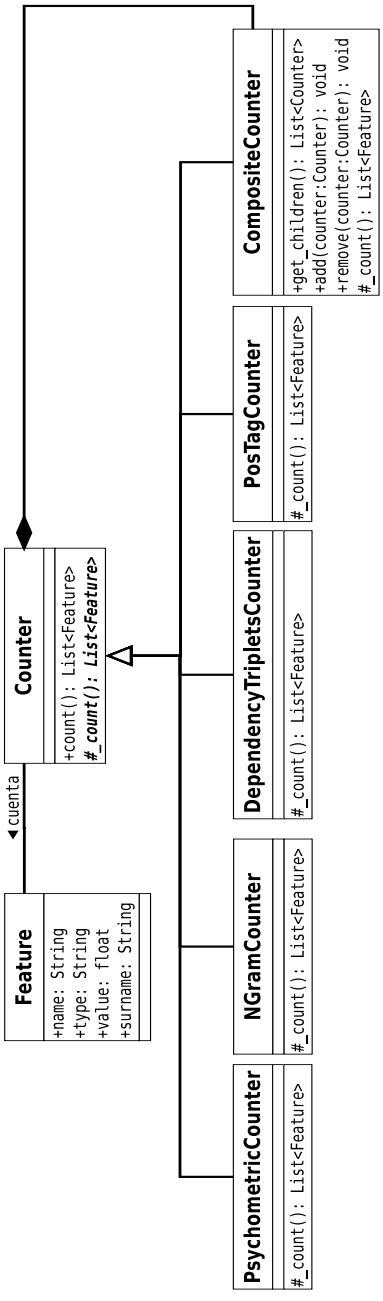

Figura 5.1: Jerarquía de clases de *Counter*

Capítulo 6

Identificación de tendencias políticas en Twitter

Este capítulo describe cómo se ha abordado la tarea de identificación de tendencias políticas de usuarios en la red social Twitter. En primer lugar se describe la naturaleza de la tarea así como los principales retos y dificultades que es necesario abordar, para a continuación presentar los modelos propuestos.

6.1. Enfoque multi-clase

En el capítulo anterior se justificó la naturaleza del enfoque multi-etiqueta en la clasificación de tópicos y se presentaron los tipos de tareas de clasificación más habituales, entre las que se encontraba la clasificación multi-clase, donde dadas tres o más clases, el clasificador debe aprender a asignar cada ejemplo de la muestra a una de ellas. Esta última es la perspectiva empleada para resolver esta tarea. El objetivo es crear un modelo que permita distinguir la ideología de usuarios que comparten opiniones políticas en la red social Twitter. Tres son las categorías consideradas en este trabajo fin de máster: *derecha*, *izquierda* y *centro*.

6.2. Retos

Uno de los principales problemas a la hora de determinar la tendencia política de usuarios es el *status quo* y el clima político en cada momento. Ello provoca que la forma en la que los políticos y sus seguidores se expresen varíe frecuentemente. Por ejemplo, aunque los términos *'paro'* y *'desempleo'* son sinónimos, su utilización puede estar sujeta a intereses políticos. Así,

es más probable que el partido del gobierno emplee la segunda opción para mitigar el impacto, mientras que la oposición optará por no utilizar expresiones eufemísticas. Expresiones como *'crecimiento negativo' versus 'recesión'* o *'ayudas' versus 'rescate'* son otros ejemplos bien conocidos. En segundo lugar, es complicado establecer un criterio aceptado unánimemente para determinar qué debe ser considerado como un pensamiento conservador o progresista, ya que dicha percepción es subjetiva para cada individuo, dependiendo de aspectos como cultura, edad o situación geográfica. Todo ello convierte el problema a resolver en una tarea con un alto nivel de abstracción.

En relación al castellano, no tenemos conocimiento de que existan conjuntos de entrenamiento que permitan crear modelos supervisados. Por ello, como parte de este trabajo fin de máster se ha creado una colección de tuits con contenido político, que permita entrenar nuestro modelo. Se utilizó la API REST de Twitter para descargar un conjunto de tuits con ciertas características. En particular se descargaron tuits de las cuentas de los cuatro principales partidos (PP, PSOE, IU y UPYD), así como de sus máximos dirigentes. En particular, se obtuvieron más de 25 367 tuits de 16 cuentas asociadas con el PP, 28 180 de 11 usuarios provenientes del PSOE, 28 418 desde nueve cuentas de IU y 18 953 desde seis políticos de UPYD. Para descargar todos estos tuits se utilizó la API de Twitter y en concreto, el método que permite recuperar el *timeline* de un usuario. Este método únicamente permite descargar los 3 200 tuits más recientes de un usuario. Ello puede suponer un problema a la hora de evaluar nuestro modelo, dado que los tuits del corpus TASS 2013 son del año 2012, como se comentó en capítulos anteriores. La solución ideal sería de disponer de mensajes del mismo espacio temporal, para que el modelo pueda ajustarse a las circunstancias de esa época. Sin embargo, las restricciones de la API de Twitter no nos han permitido recolectar mensajes de ese mismo período de tiempo. Otro punto controvertido es el de asignar a cada uno de los usuarios una tendencia política. Esta discrepancia se manifiesta a través de distintos sondeos y encuestas. En concreto, el PSOE se define a sí mismo en su manifiesto oficial como un partido de izquierdas, aunque los ciudadanos lo situaron como de centro en el barómetro del CIS de 2012 [13]. Por ello, se decidió crear dos versiones del corpus: la que sigue los criterios del CIS (con PP a la derecha, IU a la izquierda y UPYD y PSOE en el centro) y la que se basa en los manifiestos de los partidos (con IU y PSOE a la izquierda, UPYD en el centro y el PP a la derecha).

6.3. Modelos propuestos

Los modelos propuestos y sus características se corresponden con algunos de los modelos básicos descritos en capítulos anteriores. En concreto el enfoque propuesto se basa en los modelos de ngramas y psicométrico. Nuestra hipótesis asume que la palabras proporcionan en este caso un alto valor en términos de información, sin que la información morfosintáctica aporte nuevo conocimiento que ayude a mejorar el rendimiento. Dado que el corpus creado para entrenar nuestro sistema presenta dos versiones distintas en términos de anotación, se crearon dos clasificadores distintos: un primer modelo donde los tuits correspondientes a personalidades del PSOE son considerados como de centro y un segundo sistema donde serán clasificados como de izquierda.

Capítulo 7

Resultados experimentales

En este capítulo se presentan los resultados obtenidos por las aproximaciones presentadas en este trabajo fin de máster, tanto para la tarea de clasificación de tópicos como para la identificación de la tendencia política de usuarios. Se describen las métricas utilizadas para medir el rendimiento en cada una de las tareas para a continuación ilustrar y discutir los resultados para los dos retos propuestos.

7.1. Métricas de evaluación

Las métricas utilizadas para evaluar las tareas de clasificación de temas en mensajes de Twitter y detección de la tendencia política de usuarios de esta misma red social son distintas, dada la diferente naturaleza de cada una de ellas. La primera es una tarea de clasificación multi-etiqueta donde a una misma instancia (en este caso mensajes) se le puede asignar más de una clase, mientras que la detección de la tendencia política se ha abordado como un problema multi-clase donde sólo es posible asignar una categoría.

7.1.1. Clasificación multi-etiqueta: identificación de tópicos

Se utilizan las métricas estándar en el entorno de clasificación multi-etiqueta para evaluar la tarea de identificación de tópicos: la *Hamming loss* (HL), la *label-based accuracy* y la *exact match*, calculadas conforme a las ecuaciones 7.1, 7.2 y 7.3, donde:

- L es el conjunto de etiquetas.
- D es el conjunto de instancias de la colección.
- Y_i es el conjunto de etiquetas esperadas para una instancia i.

- Z_i es el conjunto de las etiquetas predichas para una instancia i.
- $\triangle$ representa la operación de diferencia simétrica entre conjuntos.

$$Hamming\ loss = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{|Y_i \triangle Z_i|}{|L|} \tag{7.1}$$

$$Label\text{–}based\ accuracy = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{|Y_i \cap Z_i|}{|Y_i \cup Z_i|} \tag{7.2}$$

$$Exact\ match = \frac{\#instancias\ clasificadas\ correctamente}{\#instancias} \tag{7.3}$$

Estas métricas permiten medir diferentes aspectos cuya relevancia debería depender del tipo de aplicación en el que se vaya a desplegar el sistema. El ejemplo 7.1.1 ilustra el comportamiento de las tres medidas propuestas.

Ejemplo 7.1.1. Dados dos tuits, $t1$ y $t2$, donde:

- $t1_a = \{$*política, economía*$\}$, representa los tópicos esperados para $t1$.
- $t1_p = \{$*deportes, economía*$\}$ representa los tópicos predichos para $t1$.
- $t2_a = \{$*deportes, películas, entretenimiento, fútbol, economía*$\}$ se refiere a los tópicos esperados para $t2$.
- $t2_p = \{$*política, películas, entretenimiento, fútbol,economía*$\}$ se corresponde con los tópicos predichos para $t2$.

La Hamming loss mide la relación del número de etiquetas erróneas con respecto al número total de etiquetas esperadas. Se trata de una función de pérdida por lo que el valor óptimo es cero. El gran inconveniente de esta métrica es su incapacidad para reflejar apropiadamente el porcentaje de etiquetas acertadas. Si calculamos la Hamming loss para $t1$ y $t2$, obtendríamos:

$$HL_{t1} = \frac{|t1_a \triangle t1_p|}{|L|} = \frac{|\{deportes,\ política\}|}{|L|} = \frac{2}{|L|}$$

$$HL_{t2} = \frac{|t2_a \triangle t2_p|}{|L|} = \frac{|\{deportes,\ política\}|}{|L|} = \frac{2}{|L|}$$

donde $HL_{t1} = HL_{t2}$, aunque $t2$ tiene un mayor porcentaje de etiquetas acertadas.

Por contra, la label-based accuracy es una medida que sí es capaz de armonizar el número de tópicos que no han sido asignados con respecto a los que han sido predichos erróneamente. En este caso, calculando la LBA para $t1$ y $t2$ tendríamos:

$$LBA_{t1} = \frac{|t1_a \cap t1_p|}{|t1_a \cup t1_p|} = \frac{|\{economía\}|}{|\{política,\ deportes,\ economía\}|} = \frac{1}{3}$$

$$LBA_{t2} = \frac{|t2_a \cap t2_p|}{|t2_a \cup t2_p|} = \frac{|\{películas,\ entretenimiento,\ fútbol,\ economía\}|}{|\{deportes,\ política,\ películas,\ entretenimiento,\ fútbol,\ economía\}|} = \frac{2}{3}$$

concluyendo que la LBA para $t2$ es mejor que para $t1$.

Por último la exact match se corresponde con un caso especial de la LBA, donde una clasificación multi-etiqueta para una instancia i solo se considera correcta cuando $Y_i = Z_i$[1], es decir, cuando no se han producido ni falsos negativos ni falsos positivos. En este ejemplo $EM(t1) = EM(t2) = 0$. Es importante señalar que una instancia i para la que se obtenga un exact match igual a 1, tendría una label-based accuracy y una Hamming loss de 1 y 0, respectivamente.

Adicionalmente, se proporcionan resultados siguiendo el criterio oficial de los organizadores del TASS, según las ecuaciones 7.4 y 7.5: la medida *at least one* es una medida relajada que asume como válida una clasificación cuando una de las etiquetas esperadas es predicha, mientras que la medida *match all* considera válida una clasificación si todas las etiquetas esperadas son predichas, independientemente de que se produzcan falsos positivos.

$$\begin{aligned} At\ least\ one &= \frac{1}{|D|}\sum_{i=1}^{|D|} f(i) \\ \text{donde } f(i) &= \begin{cases} 1 & \text{si } Y_i \cap Z_i \neq \varnothing \\ 0 & \text{si } Y_i \cap Z_i = \varnothing \end{cases} \end{aligned} \tag{7.4}$$

$$\begin{aligned} Match\ all &= \frac{1}{|D|}\sum_{i=1}^{|D|} g(i) \\ \text{donde } g(i) &= \begin{cases} 1 & \text{si } Y_i \subseteq Z_i \\ 0 & \text{si } Y_i \nsubseteq Z_i \end{cases} \end{aligned} \tag{7.5}$$

Estas dos últimas métricas presentan sin embargo un inconveniente. Es posible obtener un resultado perfecto asignando todas las posibles etiquetas a cada tuit, por lo que son menos robustas ante un funcionamiento incorrecto del sistema.

■

[1] $Y_i = Z_i \Leftrightarrow Y_i \cap Z_i = Y_i \cup Z_i$.

7.1.2. Clasificación multiclase: identificación de la tendencia política

Se emplean las métricas estándar en el ámbito clasificación multi-clase: *precisión* (P), *recall* (R), *la medida F* (F) la *accuracy*, calculadas conforme a las ecuaciones 7.6, 7.7, 7.8 donde:

- TP indica el número de instancias asignadas correctamente a una categoría en concreto, esto es, los *verdaderos positivos*.
- FP indica el número de *falsos positivos* para una categoría, es decir aquellas instancias que han sido asignadas a una clase, pero que en realidad pertenecen a otra.
- FN indica el número de instancias que no han sido asignadas a una determinada clase, pero que sí deberían haberlo sido, esto es, los *falsos negativos*.
- TN indica el número de instancias que no han sido asignadas a una clase y que realmente no pertenecen a ella, esto es, los verdaderos negativos.

$$P = \frac{TP}{TP+FP} \tag{7.6}$$

$$R = \frac{TP}{TP+FN} \tag{7.7}$$

$$F = 2 \times \frac{Precision \times Recall}{Precision+Recall} \tag{7.8}$$

Adicionalmente, es habitual emplear la *accuracy* para medir el rendimiento global del sistema:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{7.9}$$

Como en el caso de la clasificación multi-etiqueta, estas métricas tienen diferentes propósitos y su relevancia dependerá de las capacidades que se deseen proporcionar al sistema. El ejemplo 7.1.2 ilustra la validez de estas medidas.

Ejemplo 7.1.2. Dado una tarea de clasificación en tres clases: A,B y C, se han obtenido para un sistema las siguientes estadísticas:

Clase	**#instancias totales**	**TP**	**FN**	**FP**
A	1	1	0	4
B	5	3	2	0
C	15	7	3	1

Si tomásemos como referencia la precisión, obtendríamos:

$$P(A) = \frac{1}{1+4} = \frac{1}{5}$$

$$P(B) = \frac{3}{3} = 1$$

$$P(C) = \frac{7}{7+1} = \frac{7}{8}$$

Concluyendo que la clase B es para la que mejor rendimiento obtiene el sistema. Sin embargo, la precisión permite medir cuántas de las instancias asignadas a una clase pertenecen realmente a esa clase, pero no captura qué porcentaje de elementos de una categoría se han dejado de asignar. En este ejemplo se observa que, para la clase B, dos de sus instancias no fueron clasificadas como tal (falsos negativos), y sin embargo la precisión no se ve afectada por ello.

El recall sí permite medir este aspecto, reflejando la cobertura que el sistema proporciona en cada categoría:

$$R(A) = \frac{1}{1} = 1$$

$$R(B) = \frac{3}{3+2} = \frac{3}{5}$$

$$R(C) = \frac{7}{7+3} = \frac{7}{10}$$

Concluyendo que en términos de cobertura, el sistema trata mejor la clase A.

La medida F constituye una medida armónica entre la precisión y el recall, cubriendo las carencias que estas dos medidas presentan por separado. Continuando con nuestro ejemplo:

$$F(A) = 2 \times \frac{\frac{1}{5} \times 1}{\frac{1}{5}+1} = 2 \times \frac{\frac{1}{5}}{\frac{6}{5}} = \frac{1}{3}$$

$$F(B) = 2 \times \frac{1 \times \frac{3}{5}}{1+\frac{3}{5}} = 2 \times \frac{\frac{3}{5}}{\frac{8}{5}} = \frac{3}{4}$$

$$F(C) = 2 \times \frac{\frac{7}{8} \times \frac{7}{10}}{\frac{7}{8}+\frac{7}{10}} = 2 \times \frac{\frac{49}{80}}{\frac{126}{80}} = \frac{7}{9}$$

Obteniendo que la clase que mejor resuelve el sistema en conjunto es realmente la C.

■

7.2. Corpus de entrenamiento y evaluación

Existe un procedimiento estándar para entrenar y evaluar un clasificador supervisado: (1) se emplea un conjunto de entrenamiento para construir un modelo, (2) a continuación un segundo conjunto de desarrollo sirve como referencia para ajustar los parámetros del clasificador y por último, (3) un conjunto de test es empleado para evaluar el rendimiento del modelo en un entorno real. En este capítulo, los resultados experimentales se ilustran únicamente sobre el conjunto de test, para hacerlos comparables con el resto de técnicas actuales. En el capítulo 3 introdujimos la colección que servirá para entrenar y evaluar nuestras propuestas: el corpus TASS 2013 compuesto por tuits escritos en español por distintas personalidades públicas. Cada uno de los tuits se encontraba anotado con cada una de las temáticas que en él se tratan, así como la tendencia política para cada uno de los autores de dichos tuits.

En relación al corpus de entrenamiento para la tarea de clasificación de tópicos existe un problema: el desbalanceo entre categorías. Como se observa en el Apéndice A, las clases de *entretenimiento*, *otros* y *política* son mayoritarias. Ello puede producir que el clasificador no aprenda a diferenciar correctamente entre categorías: si existen pocos ejemplos para una clase determinada es probable que el clasificador opte por 'ignorar' dicha clase con el fin de optimizar su rendimiento. No obstante, existen técnicas para adaptar el corpus para evitar dicho problema, como el *oversampling*. Dado un conjunto de clases $\{C_1C_2...C_{n-1}C_n\}$, donde n_i indica el tamaño de de la muestra para la clase i con $i \in \{1, n\}$, se incrementa el número de ejemplos de n_i hasta el $max(\{n_1n_2..n_{n-1}n_n\})$ mediante la replicación de muestras para la categoría i. Así, el clasificador dispone de número de ejemplos suficientes para poder discriminar entre todas las clases. El principal inconveniente reside en la escasez de la variedad de las muestras para las clases minoritarias, dado que los ejemplos, al ser replicados, probablemente no recojan todas las características que presenta la categoría en cuestión. Existen mecanismos similares, como el *downsampling*, donde se eliminan muestras de las clases mayoritarias para igualar el número de ejemplos al de la minoritaria. El principal inconveniente de este enfoque es la pérdida de ejemplos respecto al conjunto original, lo que puede traducirse en un descenso en el rendimiento, en especial si el conjunto original es pequeño. En este caso, se ha optado por *oversampling* dado que el conjunto de entrenamiento no es demasiado extenso y así evitar mayores pérdidas.

En relación a la identificación de tendencias políticas, el corpus TASS 2013 proporciona para cada usuario la suya, pero todos ellos forman parte del conjunto de test, no se incluye

ningún tipo de conjunto de entrenamiento. Para entrenar nuestro modelo fue necesario descargar mensajes desde la cuenta oficial de Twitter de diversos representantes de los cuatro partidos políticos más votados en nuestro país: PP, PSOE, IU y UPYD, que fueron anotados según dos criterios distintos, como se comentó en capítulos anteriores.

7.3. Experimentos

Seguidamente se muestran, en distintas subsecciones, los resultados experimentales para las dos tareas propuestas en este trabajo fin de máster.

7.3.1. Clasificación de tópicos

En la tabla 7.1 se ilustran los resultados para el conjunto de los modelos base propuestos en capítulos anteriores. La información se encuentra ordenada en función de la métrica *exact match*, en orden descendente. Los bigramas de lemas obtienen el mejor rendimiento tomando como referencia la métrica *exact-match*, mientras que el modelo de palabras consigue mejores valores para las métricas más relajadas: la *Hamming loss* y la *label-based accuracy*. En los dos casos se ha aplicado previamente un filtro de ganancia de información.

La información morfológica no es de utilidad por sí misma, siendo tanto el valor de la *exact match* como la *label-based accuracy* muy bajos, confirmando una hipótesis que ya había sido postulada en capítulos anteriores. Destaca también el pobre rendimiento de la información psicométrica, incluso por debajo del modelo basado de información morfológica de grano fino. Ello pone de manifiesto uno de los inconvenientes de los recursos construidos manualmente, en términos de cobertura, especialmente en entornos web, donde el uso de términos no reconocidos es habitual. Los modelos basados en ngramas sí obtienen un buen rendimiento de partida. Destaca el aumento de rendimiento tanto para el modelo de unigramas de lemas como de palabras, cuando se aplican filtros de ganancia de información (IG) para seleccionar únicamente aquellas características que aportan ganancia (IG > 0), reforzando la necesidad de aplicar este tipo de técnicas cuando el número de características es alto. Más concretamente, los resultados para las tres métricas estándar en clasificación multi-etiqueta son mejores en el modelo de unigramas que en el de lemas. Sin embargo, la tendencia parece no mantenerse cuando se emplean ngramas de longitud mayor que uno, debido en parte al aumento de la dispersión. El espacio dimensional aumenta significativamente y el modelo no es capaz de aprender correctamente el significado de una gran cantidad de ngramas, que aparecen

muy pocas veces en el conjunto de entrenamiento. En estos casos, sí es conveniente tratar de aplicar técnicas de normalización sobre los términos, como se observa en la tabla 7.1, donde los ngramas son lematizados. En este trabajo fin de máster únicamente se ha considerado esta técnica, aunque existen otras, como el *stemming*, [2] que podrían ser incorporadas en el futuro.

Model	**IG**	**HL**	**LBA**	**EM**
Bigramas de lemas (BL)	Sí	0.077	0.626	**0.530**
Palabras (W)	Sí	**0.073**	**0.658**	0.527
Bigramas de palabras (BW)	Sí	0.080	0.613	0.524
Palabras (W	No	0.079	0.634	0.498
Lemas (L)	Sí	0.078	0.640	0.493
Lemas (L)	No	0.085	0.611	0.460
Información morfológica (grano fino) (FT)	Sí	0.289	0.262	0.032
Propiedades psicométricas (P)	Sí	0.301	0.250	0.026
Información morfológica (grano grueso) (CT)	No	0.384	0.186	0.003

Tabla 7.1: Rendimiento para los modelos de características iniciales.

Por su parte, la tabla 7.2 ilustra el rendimiento de varios modelos que combinan información, desde un punto de vista léxico, sin emplear ningún tipo de información sintáctica. Se emplean como modelos base aquellos que obtuvieron el mejor valor para alguna de las métricas estándar, esto es, los modelos de bigramas de lemas y de unigramas de palabras. Los resultados sugieren que combinar modelos de ngramas con información morfológica y psicométrica no logra incrementar significativamente el rendimiento, en la mayor parte de los casos. Por otro lado, combinar los dos mejores modelos iniciales según las métricas estándar si mejora el rendimiento. Ello refuerza la hipótesis de que combinar conocimiento léxico con información contextual permite obtener modelos más precisos.

La tabla 7.3 ilustra el rendimiento cuando al modelo de bolsa de palabras, se le incorpora información contextual mediante tripletas de dependencias, en lugar de ngramas estándar. El modelo que agrega la tripleta sintáctica no generalizada mejora ligeramente el rendimiento de su correspondiente versión léxica. Las tripletas generalizadas también mejoran el rendimiento del modelo base. El modelo constituido por palabras y tripletas de lemas donde el término padre es eliminado, mejora a su homólogo léxico formado por palabras y lemas. Nuestra

[2] Se denomina *stemming* al proceso que permite obtener la raíz de un término. Así, la raíz de las palabras *'jugamos'* y *'jugáis'* sería en ambos casos *'jug-'*.

Modelo	HL	LBA	EM
W+BL	**0.068**	**0.671**	**0.573**
BL+P	0.076	0.632	0.539
BL	0.077	0.626	0.530
W+BW+P	0.078	0.647	0.530
W+BW	0.074	0.646	0.529
W+P+FT+DT	0.073	0.655	0.527
W+P+FT	0.073	0.656	0.528
W	0.073	0.658	0.527
W+P	0.073	0.655	0.526
BL+P+FT+DT	0.081	0.615	0.498
BL+P+FT	0.082	0.612	0.495

Tabla 7.2: Rendimiento al combinar conjunto de características iniciales: *bigramas de lemas* (BL), *bigramas de palabras* (BW), *propiedades psicométricas (P)*, *palabras* (W), *lemas* (L), *etiquetas de grano fino* (FT), *tipo de dependencia* (DT)

hipótesis es que palabras marcadas con funciones sintácticas importantes, como *atributo* o *complemento directo*, pueden ser relevantes para identificar los núcleos del mensaje, y por tanto sus tópicos.

Por su parte, la tabla 7.4 compara el rendimiento de nuestra propuesta en el marco del TASS 2013, el corpus de evaluación estándar para el análisis de sistemas de clasificación de tópicos en castellano. Los resultados confirman la validez de nuestro modelo, que mejora el rendimiento del resto de aproximaciones. Aproximaciones como la propuesta por el grupo del FHC25-IMDEA obtienen un rendimiento cercano al nuestro. No obstante, la técnica de FHC25-IMDEA no aborda el problema desde una perspectiva multi-etiqueta, sino multi-clase, lo que

Características	HL	LBA	EM
W	0.073	0.658	0.527
W+(_,DT,L)	0.071	0.66	0.542
W+(L,DT,P)	0.071	0.661	0.551
W+(L,DT,L)	**0.067**	**0.674**	**0.579**

Tabla 7.3: Rendimiento al incorporar características sintácticas sobre el modelo de bolsa de palabras: *palabras* (W), *lemas* (L), *tipo de dependencia* (DT) y *propiedades psicométricas* (P)

provoca que las métricas *at least one* y *match all* sean iguales. En este corpus en concreto, un enfoque como este tiene ventaja, dada la distribución de frecuencia de los tuits (ver Ápendice A).

Model	HL	LBA	EM	Match all	At least one
Enfoque de este TFM	**0.068**	**0.674**	**0.579**	**0.663**	**0.771**
FHC25-IMDEA [14]	0.072	0.637	0.573	0.573	0.702
UPV [32]	0.084	0.608	0.468	0.659	0.756
UNED-JRM [34]	0.124	0.417	0.358	0.382	0.479
ETH-ZURICH [18]	0.098	0.370	0.291	0.385	0.455
LSI UNED [22]	0.185	0.197	0.070	0.364	0.406
SINAI-CESA [26]	0.182	0.126	0.093	0.093	0.159

Tabla 7.4: Comparación del mejor modelo propuesto en este trabajo fin de máster con el de otros métodos propuestos en el TASS 2013. Los métodos se encuentran ordenados en función de su *label-based accuracy*, en orden descendente. Algunos grupos enviaron varios experimentos, aunque nosotros solo ilustramos en este tabla el modelo con el que obtuvieron los mejores valores para las métricas estándar en clasificación multi-etiqueta.

Por último la tabla 7.5 ilustra el rendimiento para nuestro mejor clasificador y para el modelo base (únicamente palabras aplicando un filtro de ganancia de información). Es importante remarcar que tanto la *precisión*, como el *recall* o la *medida F* no son métricas pensadas para evaluar sistemas de clasificación multi-etiqueta, pero permiten obtener información relevante sobre como se comporta el sistema en cada una de las categorías. La precisión es mayor en los modelos que relacionan términos estructuralmente, pero esto mismo no se cumple en lo referido al *recall*. Ello sugiere que la información estructural permite discriminar mejor los tópicos no relacionados, mientras que los modelos basados en bolsas de términos, asignan muchas temáticas a cada tuit, alcanzando una amplia cobertura pero una precisión pobre, algo que probablemente no buscan las organizaciones. Estos resultados y conclusiones son además coherentes con las obtenidas con las métricas estándar para la evaluación de sistemas multi-etiqueta, comentadas sobre estas líneas.

7.3.2. Identificación de la tendencia política

El modelo evaluado se basa en unigramas de lemas y categorías psicométricas. Dado el buen rendimiento de los modelos de clasificación de tópicos que empleaban técnicas de ganancia de información, en esta tarea también se emplearon. La tablas 7.6 y 7.7 muestran

Categoría	medida F		P (precisión)		R (recall)	
	Mejor modelo	Palabras	Mejor modelo	Palabras	Mejor modelo	Palabras
cine	0.359	0.306	0.337	0.216	0.414	0.523
política	0.717	0.733	0.754	0.754	0.685	0.714
tecnología	0.343	0.344	0.3429	0.252	0.286	0.540
entretenimiento	0.434	0.458	0.443	0.335	0.412	0.650
deportes	0.271	0.271	0.349	0.224	0.222	0.341
otros	0.678	0.689	0.578	0.611	0.820	0.790
economía	0.435	0.391	0.372	0.267	0.524	0.729
música	0.435	0.436	0.361	0.436	0.559	0.710
fútbol	0.292	0.332	0.503	0.301	0.205	0.371
literatura	0.380	0.348	0.395	0.255	0.366	0.548
Media no ponderada	0.436	0.429	0.452	0.353	0.449	0.590

Tabla 7.5: Rendimiento por categorías para el mejor modelo sintáctico y el modelo base

el rendimiento por categoría para los dos modelos propuestos. Los criterios obtenidos mediante la encuesta del CIS (modelo 1) permiten obtener unos resultados más equilibrados en las tres categorías que su modelo homólogo basado en los manifiestos oficiales (modelo 2). Ello demuestra una mayor coherencia entre lo que perciben los usuarios y el discurso de los mensajes publicados por los políticos en la red social Twitter.

Categoría	medida F	P (precisión)	R (recall)
derecha	0.667	0.8	0.571
centro	0.324	0.268	0.407
izquierda	0.343	0.333	0.353

Tabla 7.6: Rendimiento para el modelo 1: creado según la opinión ciudadana en la encuesta del CIS 2012 (PSOE en el centro)

A pesar de ello, la accuracy del segundo modelo (0.524) supera a la del primero (0.476), dada la distribución del conjunto de test donde los usuarios conservadores constituyen la mayoría. A este respecto, el modelo 2 obtiene una mejor medida F para la clase *derecha*, posiblemente debido a que en este enfoque la clase *centro* es minoritaria y así el clasificador aprende a distinguir mejor entre *derecha* e *izquierda*. La tabla 7.8 ilustra la distribución de frecuencias en el conjunto de test.

Para ambos modelos, el centro es la categoría más complicada de clasificar. La complejidad de dicha categoría reside en su propia naturaleza: un usuario de centro compartirá ciertos argumentos con autores más conservadores y otros con los más progresistas, lo que sumado

Categoría	**medida F**	**P (precisión)**	**R (recall)**
derecha	0.685	0.792	0.603
centro	0.111	0.222	0.074
izquierda	0.532	0.417	0.735

Tabla 7.7: Rendimiento para el modelo 2: basado en los manifiestos de los partidos (PSOE en la izquierda)

a las posibles polarizaciones del corpus de entrenamiento y test puede causar una caída en el rendimiento del clasificador supervisado.

Categoría	**# autores**	**% autores**
derecha	63	50.4
izquierda	34	27.2
centro	33	26.4
Total	125	100

Tabla 7.8: Distribución de frecuencias en el conjunto de test

Capítulo 8

Conclusiones

En este capítulo presentamos las conclusiones sobre el trabajo desarrollado, resaltando los aspectos más destacados. También se realiza una exposición sobre las líneas futuras, enfocadas a enriquecer las técnicas aquí propuestas, así como su adaptación a otros ámbitos del análisis automático de textos web.

8.1. Conclusiones

El imparable avance de las tecnologías informáticas, la Web y las redes sociales, ha causado un cambio drástico e irreversible en el entorno al que se enfrentan los sectores industrial, comercial y de servicios. Con clientes y usuarios cada vez más informados, se hace imprescindible estar al tanto de las opiniones de éstos para descubrir sus necesidades y adaptar rápidamente la oferta a ellas. A este respecto, el desarrollo de técnicas de procesamiento de lenguaje humano es un aspecto clave para lograr que las máquinas puedan analizar y comprender cómo nos comunicamos y expresamos. Este trabajo fin de máster ha abordado tareas de análisis automático de los contenidos expresados por los usuarios en las redes sociales, y más concretamente sobre Twitter, una de las más populares en la actualidad. Diversas compañías y organizaciones se han interesado en monitorizar esta red social, muy útil desde el punto de vista de inteligencia de negocio y de marketing. Sin embargo, al tratarse de un medio ruidoso, donde millones de mensajes no relacionados con su sector de negocio son publicados, es necesario aplicar técnicas previas de filtrado. Así, este trabajo ha propuesto una serie de aproximaciones lingüísticas que permiten identificar las temáticas que están tratando un conjunto de tuits, ampliando las capacidades de filtrado a más que una lista finita de palabras clave expresadas en un consulta por el usuario. Los resultados demuestran que el

enfoque aquí propuesto mejora el estado del arte de las técnicas actuales para el castellano. Los avances aquí presentados han sido aceptados para su publicación en [43].

Este trabajo fin de máster ha tratado, además, la identificación de las tendencias políticas de usuarios web en Twitter, siguiendo también un enfoque lingüístico. Clasificar a un usuario como conservador o progresista es un tema controvertido y subjetivo. Por ello se han evaluado distintos modelos que siguen distintos criterios de clasificación. Los resultados experimentales refuerzan la complejidad de análisis de esta tarea, y la especial dificultad en discernir partidos de centro y de izquierda.

8.2. Trabajo futuro

La principal aportación del trabajo fin de máster reside en la utilización combinada de información léxica, psicométrica, sintáctica y semántica. Se ha tratado de crear nuevas técnicas de análisis automático de texto para disminuir nuestra desventaja competitiva respecto a los países de habla inglesa, para la cual el desarrollo de las aplicaciones de tecnologías del lenguaje está más avanzado. Tradicionalmente, las aproximaciones de clasificación automática de textos para el castellano, y más concretamente de clasificación de tópicos, se habían basado en la utilización de conocimiento léxico. Esta memoria ha mostrado las carencias de este tipo de aproximaciones, ilustrando cómo la inclusión de información sintáctica permite relacionar términos para mejorar el rendimiento de este tipo de sistemas.

En relación a las tareas futuras, creemos que aún existe mucho margen de mejora. Los enfoques propuestos son exclusivamente lingüísticos, sin hacer utilización de ningún tipo de meta información. Ello facilita la adaptación del sistema a otros ámbitos y medios sociales, pero a la vez puede suponer la pérdida de información útil. A este respecto, la inclusión de meta información de los usuarios de Twitter puede servir para ayudar a incrementar el rendimiento en este tipo de tareas. Datos como su descripción habitualmente permiten conocer a que se dedica el autor de un determinado mensaje. Del mismo modo, aspectos como la localización pueden arrojar ciertas pistas para discriminar ciertas temáticas que puede tratar dicho autor. Por otro lado, también nos gustaría explorar técnicas sobre como generar automáticamente lexicones que se adapten a un dominio en concreto. Las técnicas propuestas en este trabajo utilizan sus respectivos corpus de entrenamiento como fuente principal para extraer el conocimiento que permita construir los modelos. El principal inconveniente de esta aproximación reside en la imposibilidad de disponer de conocimiento que no se encuentre

presente en el conjunto de entrenamiento. La utilización de lexicones actúa como un buen complemento para disponer de una mayor información y supone un posible punto de partida para la creación de aproximaciones híbridas que cubran las carencias de los enfoques puros basados en aprendizaje automático. A este respecto, estudios como [24] pueden servir para enriquecer nuestra propuesta.

Por último, los modelos aquí propuestos deberían poder adaptarse para emplearse para otras tareas enmarcadas en el análisis de textos web, como el análisis del sentimiento o el análisis de reputación online, distinguiendo entre categorías como el *liderazgo*, el *rendimiento* o la *innovación*.

8.3. Competencias adquiridas

En el desarrollo de este trabajo fin de máster se han aplicado conocimientos adquiridos en diversas asignaturas del máster, principalmente:

- *Dirección de Proyectos*: desarrollo de las aptitudes de gestión y aplicación de las mismas a la gestión del ciclo de vida de un proyecto, de los recursos y agentes implicados, planificación, control y evaluación.
- *Análisis de Sistemas de información*: análisis, modelado, gestión y documentación de requisitos.
- *Inteligencia de negocio*: minería de datos.
- *Recuperación de la información y web semántica:* recuperación de información web, minería de opiniones y representación del conocimiento.

Además de profundizar en las competencias de la titulación que ya se adquirieron al cursar las asignaturas del plan de estudios, la realización del trabajo fin de máster ha supuesto el desarrollo pleno de la competencia AP16: *'Realización, presentación e defensa, unha vez obtidos todos os créditos do plan de estudos, dun exercicio orixinal realizado individualmente ante un tribunal universitario, consistente nun proxecto integral de enxeñaría en informática de natureza profesional en que se sinteticen as competencias adquiridas nas ensinanzas. '*

Apéndices

Apéndice A

Estadísticas del corpus TASS 2013

A.1. Conjunto de entrenamiento

Categorías	%tuits	#tuits
{cine}	1.5	107
{cine, economía}	0.0	1
{cine, entretenimiento}	0.3	21
{cine, entretenimiento, música}	0.0	1
{cine, entretenimiento, otros}	0.0	2
{cine, entretenimiento, política}	0.0	3
{cine, fútbol}	0.0	1
{cine, música}	0.1	7
{cine, otros}	1.3	97
{cine, otros, política}	0.0	1
{cine, tecnología}	0.1	4
{deportes}	1.0	75
{deportes, economía}	0.0	2
{deportes, entretenimiento}	0.2	11
{deportes, entretenimiento, música}	0.0	1
{deportes, entretenimiento, otros}	0.0	1
{deportes, entretenimiento, otros, política}	0.0	1
{deportes, entretenimiento, política}	0.0	1

Continúa en la siguiente página

Categorías	**%tuits**	**#tuits**
{deportes, fútbol}	0.1	5
{deportes, literatura}	0.0	1
{deportes, música}	0.1	4
{deportes, música, otros}	0.0	1
{deportes, otros}	0.1	8
{deportes, política}	0.0	1
{deportes, tecnología}	0.0	1
{economía}	3.7	267
{economía, entretenimiento}	0.4	32
{economía, entretenimiento, otros}	0.1	5
{economía, entretenimiento, otros, política}	0.0	2
{economía, entretenimiento, política}	0.5	36
{economía, entretenimiento, política, tecnología}	0.0	1
{economía, entretenimiento, tecnología}	0.0	2
{economía, fútbol}	0.0	2
{economía, literatura}	0.0	1
{economía, literatura, política}	0.0	1
{economía, literatura, política, tecnología}	0.0	1
{economía, música}	0.0	1
{economía, música, política}	0.0	1
{economía, otros}	0.3	23
{economía, otros, política}	0.4	28
{economía, otros, tecnología}	0.0	1
{economía, política}	7.3	529
{economía, política, tecnología}	0.0	1
{economía, tecnología}	0.1	5
{entretenimiento}	11.5	827
{entretenimiento, fútbol}	0.4	30
{entretenimiento, fútbol, música, otros}	0.0	1
{entretenimiento, fútbol, otros}	0.0	2

Continúa en la siguiente página

Categorías	**%tuits**	**#tuits**
{entretenimiento, literatura}	0.3	19
{entretenimiento, literatura, política}	0.0	1
{entretenimiento, literatura, tecnología}	0.0	2
{entretenimiento, música}	0.5	39
{entretenimiento, música, otros}	0.1	5
{entretenimiento, música, política}	0.0	1
{entretenimiento, música, tecnología}	0.0	1
{entretenimiento, otros}	4.5	328
{entretenimiento, otros, política}	0.2	13
{entretenimiento, otros, tecnología}	0.1	4
{entretenimiento, política}	3.3	241
{entretenimiento, política, tecnología}	0.1	4
{entretenimiento, tecnología}	0.6	40
{fútbol}	2.3	166
{fútbol, literatura}	0.0	1
{fútbol, música}	0.1	8
{fútbol, música, otros}	0.0	1
{fútbol, otros}	0.4	27
{fútbol, política}	0.1	7
{fútbol, tecnología}	0.0	1
{literatura}	0.6	45
{literatura, música}	0.0	2
{literatura, otros}	0.2	14
{literatura, política}	0.2	13
{literatura, tecnología}	0.0	2
{música}	2.8	200
{música, otros}	3.9	279
{música, otros, política}	0.0	1
{música, otros, tecnología}	0.0	1
{música, política}	0.1	5

Continúa en la siguiente página

Categorías	%tuits	#tuits
{música, tecnología}	0.1	6
{otros}	17.3	1 248
{otros, política}	3.0	215
{otros, política, tecnología}	0.0	1
{otros, tecnología}	0.4	27
{política}	27.5	1 982
{política, tecnología}	0.4	29
{tecnología}	1.1	83

A.2. Conjunto de test

Categorías	%tuits	#tuits
{cine}	0.3	203
{cine, entretenimiento}	0.0	13
{cine, entretenimiento, otros}	0.0	1
{cine, música}	0.0	5
{cine, otros}	0.6	368
{cine, política}	0.0	5
{cine, tecnología}	0.0	1
{deportes}	0.2	106
{deportes, entretenimiento}	0.0	3
{deportes, fútbol}	0.0	1
{deportes, música}	0.0	1
{deportes, otros}	0.0	20
{deportes, política}	0.0	4
{economía}	2.0	1 209
{economía, entretenimiento}	0.0	4
{economía, entretenimiento, política}	0.0	1

Continúa en la siguiente página

Categorías	**%tuits**	**#tuits**
{economía, fútbol}	0.0	1
{economía, otros}	0.3	195
{economía, otros, política}	0.0	1
{economía, política}	1.9	1 138
{entretenimiento}	5.7	3 494
{entretenimiento, fútbol}	0.0	6
{entretenimiento, literatura}	0.0	9
{entretenimiento, música}	0.0	6
{entretenimiento, música, otros}	0.0	1
{entretenimiento, otros}	2.4	1 486
{entretenimiento, otros, política}	0.0	3
{entretenimiento, otros, tecnología}	0.0	3
{entretenimiento, política}	0.6	371
{entretenimiento, tecnología}	0.0	20
{fútbol}	1.2	700
{fútbol, música}	0.0	2
{fútbol, otros}	0.2	95
{fútbol, política}	0.0	17
{fútbol, tecnología}	0.0	1
{literatura}	0.1	76
{literatura, otros}	0.0	7
{literatura, política}	0.0	1
{música}	0.9	545
{música, otros}	1.5	924
{música, política}	0.0	13
{música, tecnología}	0.0	1
{otros}	34.5	20 979
{otros, política}	6.7	4 081
{otros, tecnología}	0.0	27
{política}	40.2	24 416

Continúa en la siguiente página

Categorías	**%tuits**	**#tuits**
{política, tecnología}	0.0	16
{tecnología}	0.4	218

Glosario

AA: Aprendizaje automático.

API: Application Programming Interface

KLD: Kullback Leibler Divergence.

LA: Label Accuracy Score.

LAS: Labeled Attachment Score.

LDA: Latent Dirichlet Allocation.

LIWC: Linguistic Inquiry and Word Count.

NLTK: Natural Language Toolkit.

PLN: Procesamiento del Lenguaje Natural.

REST: Representational State Transfer.

UAS: Unlabeled Attachment Score.

SVM: Support Vector Machines.

Bibliografía

[1] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau. Sentiment analysis of Twitter data. In *Proceedings of the Workshop on Languages in Social Media*, LSM '11, pages 30–38, Stroudsburg, PA, USA, 2011. ACL.

[2] Jordi Atserias, Bernardino Casas, Elisabet Comelles, Meritxell González, Lluís Padró, and Muntsa Padró. Freeling 1.3: Syntactic and semantic services in an open-source nlp library. In *Proceedings of LREC*, volume 6, pages 48–55, 2006.

[3] Marco Toledo Bastos, Rodrigo Travitzki, and Cornelius Puschmann. What sticks with whom? Twitter follower-folowee networks and news classfication. In *AAAI Technical Report WS-12-01 Workshop on the Potential of Social Media Tools and Data for Journalists*, pages 6–13, Dublin, Ireland, June 2012. AAAI.

[4] Fernando Batista and Ricardo Ribeiro. The L2F strategy for sentiment analysis and topic classification. In *TASS 2012 Working Notes*, Castellón de la Plana, Spain, September 2012.

[5] Thorsten Brants. TnT: a statistical part-of-speech tagger. In *Proceedings of the sixth conference on Applied natural language processing*, ANLC '00, pages 224–231, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics.

[6] E. Brill. A simple rule-based part of speech tagger. In *Proceedings of the workshop on Speech and Natural Language*, HLT'91, pages 112–116, Stroudsburg, PA, USA, 1992. Association for Computational Linguistics.

[7] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. In *Proc. of the Seventh International Conference on World Wide Web*, pages 107–117, Brisbane, Australia, 1998.

[8] Sabine Buchholz and Erwin Marsi. Conll-x shared task on multilingual dependency parsing. In *Proceedings of the Tenth Conference on Computational Natural Language Learning*, CoNLL-X '06, pages 149–164, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[9] Stefan Büttcher, Charles L. A. Clarke, and Gordon V. Cormack. *Information Retrieval: Implementing and Evaluating Search Engines*. The MIT Press, Cambridge, Massachusetts, USA, 2010.

[10] Angel Castellano González, Juan Cigarrán Recuero, and Ana García Serrano. UNED @ TASS: Using IR techniques for topic-based sentiment analysis through divergence models. In *TASS 2012 Working Notes*, Castellón de la Plana, Spain, September 2012.

[11] Angel Castellano González, Juan Cigarrán Recuero, and Ana García Serrano. UNED LSI @ TASS 2013: Considerations about textual representation for IR based tweet classification. In Díaz Esteban et al. [17], pages 213–219.

[12] C. Chang and C. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems Technology*, 2(3):Article 27, April 2011.

[13] CIS. Barómetro autonómico (iii). Technical Report 2956, Centro de Investigaciones Sociológicas, September-October 2012.

[14] Héctor Cordobés, Antonio Fernández Anta, Luis Felipe Núñez, Fernando Pérez, Teófilo Redondo, and Agustín Santos. Técnicas basadas en grafos para la categorización de tweets por tema. In Díaz Esteban et al. [17], pages 160–166.

[15] Michael Coleman Dalvean. Who's who on the australian political left-right spectrum? Working/technical paper, Australian National University, Canberra, 2013. Open Access Research http://hdl.handle.net/1885/9755.

[16] Ovidiu Dan, Junlan Feng, and Brian D. Davison. A bootstraping approach to identifying relevant tweets for social tv. In Lada A. Adamic, Ricardo A. Baeza-Yates, and Scott Counts, editors, *Proceedings of the Fifth International Conference on Weblogs and Social Media*, Barcelona, Spain, July 2011. AAAI.

[17] A. Díaz Esteban, I. Alegría Loinaz, and J. Villena Román, editors. *XXIX Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN 2013). TASS*

2013 - Workshop on Sentiment Analysis at SEPLN 2013, Madrid, Spain, September 2013. SEPLN.

[18] David García and Mike Thelwall. Political aligment and emotional expressions in Spanish tweets. In Díaz Esteban et al. [17], pages 151–159.

[19] Abhishek Gattani, Digvijay S. Lamba, Nikesh Garera, Mitul Tiwari, Xiaoyong Chai, Sanjib Das, Sri Subramaniam, Anand Rajaraman, Venky Harinarayan, and AnHai Doan. Entity extraction, linking, classification, and tagging for social media: A Wikipedia-based approach. *Proceedings of the VLSB Endowment*, 6(11):1126–1137, August 2013.

[20] M. Joshi and C. Penstein-Rosé. Generalizing dependency features for opinion mining. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, ACLShort '09, pages 313–316, Suntec, Singapore, 2009. Association for Computational Linguistics.

[21] Kathy Lee, Diana Palsetia, Ramanathan Narayanan, Md. Mostofa Ali Patwary, Ankit Agrawal, and Alok Choudhary. Twitter trending topic classification. In *Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops (ICDMW 2011)*, pages 251–258, Vancouver, Canada, December 2011. IEEE.

[22] Tamara Martín-Wanton and Jorge Carrillo de Albornoz. UNED at TASS 2012: Polarity classification and trending topic system. In *TASS 2012 Working Notes*, Castellón de la Plana, Spain, September 2012.

[23] Eugenio Martínez-Cámara, Miguel Ángel García Cumbreras, M. Teresa Martín-Valdivia, and L. Alfonso Ureña López. SINAI en TASS 2012. *Procesamiento del Lenguaje Natural*, 50:53–60, March 2013.

[24] Saif M Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. *arXiv preprint arXiv:1308.6242*, 2013.

[25] Mahdi Mohseni, Hasan Motalebi, Behrouz Minaei-bidgoli, and Mahmoud Shokrollahi-far. A farsi part-of-speech tagger based on markov model. In *Proceedings of the 2008 ACM symposium on Applied computing*, SAC '08, pages 1588–1589, New York, NY, USA, 2008. ACM.

[26] A. Montejo-Ráez, Manuel C. Díaz Galiano, and M. García-Vega. LSA based approach to TASS 2013. In Díaz Esteban et al. [17], pages 195–199.

[27] Tony Mullen and Robert Malouf. A preliminary investigation into sentiment analysis of informal political discourse. In *AAAI 2006 Spring Symposium on Computational Approaches to Analysing Weblogs (AAAI-CAAW 2006)*, 2006.

[28] J. Nivre, J. Hall, J. Nilsson, A. Chanev, G. Eryigit, S. Kübler, S. Marinov, and E. Marsi. Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135, 2007.

[29] A. Pak and P. Paroubek. Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May 2010. European Language Resources Association (ELRA).

[30] J.W. Pennebaker, M.E. Francis, and R.J. Booth. Linguistic inquiry and word count: LIWC 2001. *Mahway: Lawrence Erlbaum Associates*, page 71, 2001.

[31] Jacob Perkins. *Python Text Processing with NLTK 2.0 Cookbook*. Packt Publishing, 2010.

[32] Ferran Pla and Lluís-F. Hurtado. ELiRF-UPV en TASS-2013: Análisis de sentimientos en Twitter. In Díaz Esteban et al. [17], pages 220–227.

[33] J. C. Platt. Advances in kernel methods. chapter Fast training of support vector machines using sequential minimal optimization, pages 185–208. MIT Press, Cambridge, MA, USA, 1999.

[34] Francisco Javier Rufo Mendo. Are really different topic classification and sentiment analysis? In Díaz Esteban et al. [17], pages 206–212.

[35] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, March 2002.

[36] Bharath Sriram, Dave Fuhry, Engin Demir, Hakan Ferhatosmanoglu, and Murat Demirbas. Short text classification in Twitter to improve information filtering. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2010)*, pages 841–842, Geneva, Switzerland, July 2010. ACM.

[37] Steven Bird, Ewan Klein, Edward Loper. *Natural Language Processing with Python*. O'REILLY, 2009.

[38] Mike Thelwall, Kevan Buckley, Georgios Paltoglou, Di Cai, and Arvid Kappas. Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology*, 61(12):2544–2558, 2010.

[39] Chanattha Thongsuk, Choochart Haruechaiyasak, and Somkid Saelee. Multi-classification of business types on Twitter based on topic model. In *The 8th Electrical Engineering/Electronics, Computer, Telecommunications and Information Technologies (ECTI) Association of Thailand — Conference 2011*, pages 508–511, Khon Kaen, Thailand, May 2011.

[40] P. D. Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 417–424, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[41] Twitter, Inc. Form S-1 Registration Statement under The Securities Act of 1933. Washington, D.C.: United States Securities and Exchange Comission, 2013.

[42] David Vilares, Miguel A. Alonso, and Carlos Gómez-Rodríguez. Supervised polarity classification of Spanish tweets based on linguistic knowledge. In *DocEng'13. Proceedings of the 13th ACM Symposium on Document Engineering*, pages 169–172, Florence, Italy, September 2013. ACM.

[43] David Vilares, Miguel A. Alonso, and Carlos Gómez-Rodríguez. Evaluación de aproximaciones lingüísticas para la asignación de temas a tuits. In *III Congreso Español de Recuperación de la Información, A Coruña, Spain, 2014*, A Coruña, Spain, June 2014.

[44] Julio Villena-Román and Janine García-Morera. TASS 2013 — workshop on sentiment analysis at SEPLN 2013: An overview. In Díaz Esteban et al. [17], pages 112–125.

[45] Julio Villena-Román, Sara Lana-Serrano, Eugenio Martínez-Cámara, and José Carlos González-Cristóbal. TASS — workshop on sentiment analysis at SEPLN. *Procesamiento del Lenguaje Natural*, 50:37–44, March 2013.

[46] Surender Reddy Yerva, Zoltán Miklós, and Karl Aberer. Entity-based classification of Twitter messages. *International Journal of Computer Science and Applications*, 9(2):88–115, 2012.

[47] L. Zhang, R. Ghosh, M. Dekhil, M. Hsu, and B. Liu. Combining lexicon-based and learning-based methods for Twitter sentiment analysis. Technical Report HPL-2011-89, HP Laboratories, Palo Alto, CA, 2011.