UNIVERSIDAD POLITÉCNICA DE VALENCIA

Departamento de Sistemas Informáticos y Computación

Reconocimiento de Formas e Inteligencia Artificial

Categorización Semi-supervisada de Documentos Usando la Web como Corpus

Tesis Doctoral

Rafael Guzmán Cabrera

Directores:

Paolo Rosso, Universidad Politécnica de Valencia, España

Manuel Montes y Gómez, INAOE, México

Valencia

Noviembre 2009

Agradecimientos

Quisiera dar mi más sincero agradecimiento a todos los que me han apoyado, animado, contribuido y colaborado en la realización de la presente tesis. En primer lugar quiero destacar todo el apoyo de parte de mis directores de tesis: Paolo Rosso y Manuel Montes y Gómez. Ambos con inagotable paciencia y dedicación me han proporcionado todo tipo de ideas, consejos y revisiones en cada una de las etapas y tareas de esta tesis.
También estoy muy agradecido con mis compañeros de la división de ingeniarías de la Universidad de Guanajuato, en especial con Miguel Torres, Oscar Ibarra y René Jaime Rivas por el apoyo y animo incondicional recibido y sin el cual hubiera sido difícil la finalización del presente trabajo. Y a todos mis compañeros del departamento de sistemas informáticos y computación de la UPV y del laboratorio de tecnologías del lenguaje del INAOE, en especial a José Manuel Gómez Soriano y Alberto Téllez con quienes compartí gran parte de este tiempo de investigación.

En la parte personal, quiero hacer un agradecimiento a Clara, mi esposa, quien siempre me ha animado para que pudiera finalizar con éxito esta tesis. A mis hijos: Adriana Hayme y Gustavo Rafael, quienes han sido los más sacrificados durante este largo tiempo dedicado al desarrollo de esta tesis. A mi mama, hermanos y sobrinos por todo lo que me han ayudado en el aspecto personal y profesional.

Quiero hacer una mención especial a mis amigos, con quienes he compartido varios de los momentos de esparcimiento y con los cuales estoy muy agradecido por el apoyo incondicional recibido y con los cuales espero seguir compartiendo experiencias, en particular a Martin, Cosme, Noé y Hamlet, gracias por todo.

La realización de este trabajo de investigación ha sido posible gracias a los apoyos recibidos por parte de la Universidad de Guanajuato y el PROMEP por su apoyo a través de la beca UGTO-121.

El tribunal de tesis ha sido formado por:

- Dr. Manuel Palomar Sanz (Universidad de Alicante)
- Dr. Paloma Martínez Fernández (Universidad Carlos III de Madrid)
- Dr. Luis Villaseñor Pineda (INAOE, México)
- Dr. Grigori Sidorov (Instituto Politécnico Nacional, México)
- Dr. Antonio Molina Marco (Universidad Politécnica de Valencia)

Resumen

La mayoría de los métodos para la categorización automática de documentos está basada en técnicas de aprendizaje supervisado y por consecuencia, tienen el problema de requerir un gran número de instancias de entrenamiento. Con la finalidad de afrontar este problema, en esta tesis se propone un nuevo método semi-supervisado para la categorización de documentos, el cual considera la extracción automática de ejemplos no etiquetados de la Web y su incorporación al conjunto de entrenamiento. Los ejemplos no etiquetados que se incorporan al conjunto de entrenamiento son seleccionados por medio de un método basado en aprendizaje automático. Este modelo incremental permite la selección sólo de los mejores ejemplos no etiquetados en cada iteración. Sin embargo, en algunos dominios esta técnica no permite mejorar la precisión de clasificación, principalmente cuando los datos etiquetados son dispersos. Esto es, entre más relación tengan los ejemplos etiquetados con la categoría a la que pertenecen, mejores resultados se obtendrán con este método. Éste es independiente del dominio y del lenguaje, su funcionamiento resulta más adecuado en aquellos escenarios en los cuales no se cuenta con suficientes instancias de entrenamiento manualmente etiquetadas. La evaluación experimental del método se llevó a cabo con tres experimentos de categorización de documentos tanto temática (utilizando colecciones con diferentes características de documentos, como son: muy pocos ejemplos de entrenamiento y un alto grado de traslape) así como no temática (tarea de atribución de autoría). Un cuarto experimento se llevó a cabo para la tarea de la desambiguación del sentido de las palabras. Los resultados obtenidos en cada uno de estos experimentos nos permiten ver la efectividad de incorporar datos no etiquetados descargados de la Web al conjunto de entrenamiento.

Resum

La majoria dels mètodes per a la categorització automàtica de documents està basat en tècniques d'aprenentatge supervisat i per conseqüentment, tenen el problema de requerir un gran nombre d'instàncies d'entrenament. Amb la finalitat d'afrontar aquest problema, en aquesta tesi es proposa un nou mètode semi-supervisat per a la categorització de documents, el qual considera l'extracció automàtica d'exemples no etiquetats de la Web i la seua incorporació al conjunt d'entrenament. Els exemples no etiquetats que s'incorporen al conjunt d'entrenament són seleccionats mitjançant un mètode basat en aprenentatge automàtic. Aquest model incremental permet la selecció només dels millors exemples no etiquetats en cada iteració. No obstant això, en alguns dominis aquesta tècnica no permet millorar la precisió de classificació, principalment quan les dades etiquetades són disperses. Açò és, quanta major relació tinguen els exemples etiquetats amb la categoria a què pertanyen, millors resultats s'obtindran amb aquest mètode. Aquest mètode és independent del domini i de l'idioma, el seu funcionament resulta més adequat en aquells escenaris en els quals no es disposa de suficients instàncies d'entrenament manualment etiquetades. L'avaluació experimental del mètode es va dur a terme amb tres experiments de categorització de documents, categorització tant temàtica (utilitzant col·leccions amb diferents característiques de documents, com són: molt pocs exemples d'entrenament i un alt grau de solapament) com no temàtica (tasca d'atribució d'autoria). També es va dur a terme un quart experiment per a la tasca de la desambiguació semàntica. Els resultats obtinguts en cada un d'aquests experiments ens permeten veure la efectivitat d'incorporar dades no etiquetades descarregades de la Web al conjunt de entrenament.

Abstract

Most of methods for automatic document categorization based on supervised learning techniques and consequently, they have the problem of requiring a large number of training instances. In order to tackle this problem, this thesis proposes a new semi-supervised method for categorizing documents, which considers the automatic extraction of unlabelled examples of the Web and its incorporation into the training set. The unlabeled examples for the training set are selected by a method based on machine learning. This incremental model only allows a selection of the best examples that are not labeled in each one of the iterations. However, in some domains this technique improves the accuracy of classification, especially when labeled data are sparse. That is, the more respect they have the examples labeled with the category to which they belong, they will get better results with this method. This method is independent of the domain and language, its operation is more appropriate in those scenarios in which there is not enough manually tagged training instances. The experimental evaluation of the method was carried out with three experiments using thematic categorization of documents (using collections of documents with different characteristics, such as: very few examples of training and a high degree of overlap) and not thematic (authorship attribution). A fourth experiment was carried out for the word sense disambiguation task. The results in each of those experiments allow us to see the effectiveness of incorporating unlabeled data downloaded from the Web to the training set.

Capítulo I

Introducción

En este capítulo se presenta la descripción de uno de los problemas de investigación abiertos que motivan el desarrollo del presente trabajo de tesis: el problema de la organización y categorización de documentos. Este problema cobra importancia significativa dada la cantidad, cada vez mayor, de documentos en formato electrónico disponibles en repositorios de información tanto públicos como privados. Eso obliga a contar con herramientas que permitan organizar dicha información para facilitar su manejo. Se presentan, además, los objetivos planteados en el desarrollo del presente proyecto, así como una breve descripción del contenido de los capítulos.

1.1 Motivación

Actualmente vivimos en la era del conocimiento. En esta era el conocimiento ha recobrado un valor que trasciende a los individuos y las organizaciones. Este hecho ha transformado en gran medida la forma de vida de las clases sociales predominantes, y también, aunque de forma más paulatina, afecta ya a comunidades cada vez más alejadas de los grandes núcleos poblacionales en todo el mundo. Esta era, identificada por vez primera en la década de los 70´s como la "sociedad del conocimiento", y que se popularizó durante los años 80´s (Aurajo, 2006), se caracteriza por la forma en la que los individuos viven e interactúan. En ella los individuos hacen uso de las tecnologías de la información y las comunicaciones con la finalidad de relacionarse a distancia, realizar transacciones de todo tipo en menor tiempo, así como tener acceso, analizar, producir y asimilar cantidades de información cada vez mayores.

Muestra de esto es el crecimiento exponencial de los repositorios de información en formatos electrónicos tanto públicos como privados, en particular aquellos que se encuentran en forma escrita. El mejor ejemplo de las fuentes públicas de información es Internet, mientras que como ejemplos de repositorios privados se encuentran las bases de conocimiento de domino específico, como las bibliotecas electrónicas médicas, o de dominio abierto, como las colecciones de las agencias de noticias, por ejemplo Reuters, EFE o Los Ángeles Times.

El uso de información almacenada en medios electrónicos, y en particular, en forma textual es ya una tarea cotidiana en una gran variedad de dominios del conocimiento humano. Esto genera una serie de dependencias entre las diversas necesidades de los usuarios y los avances en la investigación y desarrollo, tanto de metodologías como de herramientas capaces de satisfacerlas.

Por una parte, los usuarios requieren de mejores y mayores repositorios de información, mientras que, por otro lado, los mecanismos actuales de acceso a las fuentes de información, como pueden ser las máquinas de búsqueda de información, son cada vez menos adecuados para el tratamiento de tales volúmenes de información dado que dejan al usuario una tarea abrumadora: filtrar la información devuelta por dichos sistemas a fin de satisfacer su necesidad inicial de información (Salton et al., 1987). Este hecho ha motivado la investigación en diversas tareas, entre las que destacan: la recuperación de información (Riloff et al., 2005), la minería de texto (Glenn et al., 2006), la extracción de información (Patwardhan et al., 2007) la búsqueda de respuestas (Molla et al., 2007; Ferrández et al., 2008) y la categorización de textos (Sebastiani, 2005). En particular, la Categorización de Textos ha sido ampliamente estudiada y ha alcanzado resultados sorprendentes al permitir el desarrollo de sistemas capaces de distinguir el área temática o tópico de un conjunto de documentos (Duda et al., 1973; Joachims, 1998; Lewis et al., 1994; Sebastiani, 2005). La gran mayoría de los métodos desarrollados determinan la categoría del documento basándose únicamente en sus palabras, sus repeticiones y las combinaciones entre ellas (Riloff et al., 2005; Fürnkranz, 1998; Peng et al., 2004; Sebastiani, 2005), es decir, no llevan a cabo una representación semántica del documento que les permita “entenderlo”.

Las herramientas desarrolladas hasta ahora tienen la finalidad de facilitar el manejo de grandes volúmenes de información textual en formato electrónico y de esta manera optimizar tanto el tiempo como la calidad de la información proporcionada como respuesta a un usuario. Sin embargo, este problema tiende a hacerse cada vez más complejo debido al incremento continuo y exponencial de la cantidad de información disponible en formato electrónico, por ejemplo en la Web.

1.2 Descripción del problema

La cantidad excesiva de documentos en lenguaje natural disponibles en formato electrónico hace imposible su análisis. Para aprovechar eficientemente la información contenida en estos documentos, se han hecho estudios en distintas líneas de investigación como las descritas en la sección anterior. Esto con el fin de invertir el menor tiempo posible, en buscar la información deseada, de manera que el resultado de la búsqueda traiga consigo los documentos (o fragmentos de ellos) con la información más relevante a la petición realizada, evitando aquellos documentos que no contienen o contienen poca información relevante. Sin duda, una buena organización de los documentos es de gran ayuda a la hora de buscar información; de esta tarea se encarga la categorización automática de documentos (Sebastiani, 2006). Investigadores de las áreas de recuperación de información y de aprendizaje automático han fijado su atención en la categorización automática de documentos, tan es así, que muchas de las herramientas usadas en esta tarea provienen precisamente de dichas áreas.

La categorización automática de documentos es la tarea de asignar a un documento nuevo, no visto antes por el conjunto de entrenamiento, la clase o categoría dentro de un conjunto previamente definido. Un documento puede pertenecer a una sola clase, a varias clases o bien no pertenecer a ninguna clase (Joachims, 1998; Montejo et al., 2007). Cuando se lleva a cabo esta tarea usando aprendizaje automático el objetivo es entrenar sistemas de categorización utilizando un conjunto de documentos de ejemplo (documentos previamente etiquetados manualmente por un experto), para que puedan realizar la tarea de asignar la clase automáticamente a nuevos documentos.

Los documentos usados en la tarea de categorización automática de documentos son indexados usando las mismas técnicas que en recuperación de información, más aún, los documentos son comparados y la similitud entre ellos es medida usando técnicas originalmente desarrolladas en recuperación de información.

Para investigadores del área de aprendizaje automático, la categorización automática de documentos es una tarea con la que pueden comparar sus técnicas y metodologías, ya que las aplicaciones de la categorización automática de documentos usan espacios de atributos de alta dimensionalidad debido a que la cantidad de datos que se maneja es muy grande.

Para desarrolladores de la industria, la categorización automática de documentos es importante por la cantidad enorme de documentos que se necesita procesar y clasificar, de manera correcta y eficiente. Pero aún más importante, es el hecho de que las técnicas de categorización automática de documentos han alcanzado niveles de exactitud comparables al desempeño de profesionales entrenados (Riloff et al., 2005; Sebastiani, 2006, Montejo et al., 2007), sin embargo, el tiempo invertido por un sistema de categorización automática de documentos es muchísimo menor que el de una persona experta.

Las técnicas de categorización automática de documentos son usadas en una gran variedad de tareas, pero en general se puede hablar de dos tipos: la categorización temática y la categorización no temática. Como ejemplo del problema de categorización temática se tiene la categorización de noticias en la redacción de un diario, donde estas se deben ubicar de acuerdo con la sección en la que aparecerán: deportes, sociales, policiaca y política, entre otras; mientras que para el caso de la categorización no temática se tiene el caso de la atribución de autoría, en esta tarea, los documentos se deben ubicar de acuerdo con el estilo de escritura, más que por las palabras que componen al documento.

En general, para que un sistema de categorización pueda llevar a cabo su tarea de una manera correcta con nuevos documentos, es necesario entrenarlo con algunos documentos previamente etiquetados de cada clase, de esa manera el sistema de categorización podrá, a partir del modelo aprendido con los documentos previamente etiquetados, generalizarlo para documentos nuevos.

Se han estudiado varios tipos de sistemas de categorización en el área de aprendizaje automático y algunos de ellos pueden ser usados para la categorización automática de documentos. Entre estos podemos mencionar a los probabilísticos (Di Nunzio, 2009), los basados en Kernels (Wang et al., 2008) y los basados en prototipos (Yang et al., 1999), los cuales se relacionan de manera más directa con trabajos hechos en recuperación de información (Riloff et al., 2005).

Un punto importante a considerar es el hecho de que el sistema de categorización debe ser eficiente e independiente del dominio de los documentos a ser categorizados. Por otro lado, los sistemas tradicionales de categorización automática de documentos, como es el caso de los sistemas supervisados, permiten obtener valores de exactitud muy buenos, sin embargo, tienen la desventaja de requerir una cantidad de instancias de entrenamiento muy grande (Di Nunzio, 2009). Estas instancias de entrenamiento algunas veces son muy difíciles de obtener para todas o algunas de las clases, ya que en la tarea de categorización automática de documentos se presenta de manera muy frecuente el problema de las clases desbalanceadas (Eibe et al., 2006), esto es, se tiene un número significativamente mayor de instancias de entrenamiento para algunas clases y muy pocos para otras, haciendo inoperantes a estos sistemas de categorización, lo que se refleja en una exactitud de categorización muy baja.

En este escenario el diseño y desarrollo de métodos de categorización que permitan el funcionamiento utilizando muy pocas instancias de entrenamiento son necesarios y en este punto es que se encuentra el presente trabajo de tesis. En particular en el uso de un método semi-supervisado basado en la Web. Los métodos semi-supervisados, como se detallará más adelante en este trabajo, tienen la ventaja de permitir el uso de pocas instancias etiquetadas de entrenamiento originalmente y después, por medio de técnicas de aprendizaje automático, incorporar instancias no etiquetadas al conjunto de entrenamiento. Este es un hecho que ha potenciado el desarrollo de este tipo de métodos, dada la facilidad que se tiene para conseguir instancias no etiquetadas por ejemplo en la Web.

1.3 Objetivos de la tesis

La presente tesis tiene como objetivo principal el desarrollo de un método semi-supervisado de categorización automática de documentos que permita la incorporación de información no etiquetada, proveniente de la Web, al conjunto de entrenamiento. El método desarrollado debe tener, además, las siguientes características:

1. *Adecuado para funcionar con muy pocos ejemplos de entrenamiento*.- Dado que, frecuentemente, es difícil contar con un número grande de instancias de entrenamiento para todas las clases en los problemas de categorización automática de documentos, la incorporación de instancias no etiquetadas de la Web al conjunto de entrenamiento puede ayudar a disminuir este problema, para lo cual se requieren pocos ejemplos de entrenamiento.

2. *No requiere de un conjunto de datos no etiquetados.*- A diferencia de otros métodos, no requiere de un conjunto previo de ejemplos no etiquetados, ya que estos son obtenidos de manera automática de la Web. Esta es una característica muy importante, ya que es común en los sistemas de categorización de documentos que utilizan datos no etiquetados el requerir un conjunto de datos del mismo dominio o con la misma estructura, lo cual no ocurre con el método propuesto en el presente trabajo.

3. *Independiente del lenguaje*.- Esta característica permitirá la aplicación del método de categorización automática de documentos en distintos idiomas, aprovechando que la Web es la fuente de los ejemplos no etiquetados que serán incorporados al conjunto de entrenamiento.

4. *Independiente del dominio*.- Esta característica permitirá llevar a cabo diversas tareas de categorización de documentos, tanto temática como no temática, siempre y cuando se cuente con instancias de entrenamiento para cada categoría.

Esta característica permite abordar, con el mismo método, diferentes problemáticas de categorización automática de documentos ya que por ejemplo, se podrá realizar la categorización de noticias acerca de desastres naturales por un lado, y por otro poder identificar a qué autor corresponde un determinado documento, los cuales intuitivamente podemos ubicar como problemáticas muy distintas.

1.4 Organización de la tesis

El presente documento está organizado de la siguiente manera: en el capítulo dos se presentan conceptos básicos del aprendizaje computacional. Estos conceptos permitirán entender mejor el contenido de la tesis, los cuales incluyen, principalmente, conocimientos de aprendizaje automático relacionados a nuestro problema de investigación así como una descripción de los algoritmos de aprendizaje utilizados en el presente trabajo. Además, se incluye una sección de categorización automática de documentos en la que se abordan conceptos clave de la arquitectura propuesta para la extracción de información.

En el capítulo tres se presenta una descripción de los principales métodos utilizados para llevar a cabo la tarea de categorización automática de documentos, como son los ensambles de clasificadores, así como diferentes enfoques de categorización semi-supervisada. Cabe mencionar que no se pretende hacer una revisión exhaustiva, pero sí una descripción de los principales trabajos que aplican técnicas similares a nuestro enfoque. Se presenta además una sección que describe el uso de la Web como corpus en diversas tareas del procesamiento del lenguaje natural, y específicamente, para la obtención de ejemplos de entrenamiento adicionales en la tarea de categorización automática de documentos. En este capítulo se muestra también uno de los problemas más comunes en tareas de categorización textual: el de las clases desbalanceadas, se hace, también, un análisis de las principales estrategias de solución a este problema. Se termina este capítulo definiendo lo correspondiente a la categorización no temática de documentos, los principales enfoques y tareas utilizados en esta área.

En el capítulo cuatro se presenta la arquitectura general del método propuesto, el cual realiza la categorización automática de documentos usando la Web como corpus. Además, se presenta una descripción de cada una de las partes que lo componen así, como las principales aportaciones del método.

En el capítulo cinco se presentan los resultados obtenidos de la aplicación del método en tres diferentes experimentos de categorización textual: categorización de noticias sobre desastres naturales en español, categorización de noticias de la colección Reuters en inglés y la atribución de autoría de poemas correspondientes a cinco poetas contemporáneos mexicanos. Para cada uno de estos experimentos se lleva a cabo una descripción del objetivo del experimento, del corpus utilizado y de los resultados obtenidos, tanto de referencia como al aplicar el método propuesto.

En el capítulo seis se presentan los resultados obtenidos al aplicar el método a la tarea de desambiguación del sentido de las palabras; esta tarea se lleva a cabo como un problema de categorización de documentos; al igual que en los casos anteriores, se presenta el objetivo del experimento, una descripción del conjunto de datos utilizados, así como los resultados obtenidos.

Finalmente, en el capítulo siete se resumen las principales conclusiones de la investigación, así como las aportaciones y las posibles líneas para trabajo futuro. Además se presenta un listado con las publicaciones logradas con el desarrollo de la presente investigación.

1.5 Conclusiones del capítulo

En este capítulo se presentó el área de investigación abierta que motiva el desarrollo de la presente investigación; se muestra una descripción del problema que deja clara la necesidad de contar con sistemas de categorización que puedan funcionar con pocos ejemplos etiquetados como parte del conjunto de entrenamiento y mejor aún, que se puedan incorporar ejemplos no etiquetados. Se resalta, también, la importancia de contar con sistemas de categorización que sean independientes tanto del lenguaje como del dominio de aplicación. En este capítulo, además, se presentaron los objetivos del presente trabajo de investigación, así como una breve descripción del contenido de cada uno de los capítulos.

Capítulo II

Categorización de documentos

En este capítulo se hace una introducción a la teoría que fundamenta la presente tesis. Se presentan las definiciones formales usadas a través del documento, las cuales pretenden, en primer lugar, familiarizar al lector con el área de la categorización automática de documentos, así como con los algoritmos y técnicas empleados para llevar a cabo esta tarea.

2.1 Aprendizaje automático

Las técnicas de aprendizaje computacional se han utilizado frecuentemente para resolver problemas donde se manejan grandes cantidades de información y es necesario encontrar un patrón que permita determinar el comportamiento dicha información. El objetivo del aprendizaje computacional es desarrollar modelos que sean capaces de aprender de la experiencia previa de los eventos que se presentan, a partir de conjuntos de datos (Aurajo, 2006). La finalidad de los modelos es extraer información implícita dentro de los datos para poder hacer predicciones y tomar decisiones sobre nuevos datos.

Una definición comúnmente utilizada es la siguiente: "*El aprendizaje automático es el estudio de algoritmos computacionales que van mejorando automáticamente su desempeño a través de la experiencia*" (Mitchell, 1997). De manera más formal, esta definición se puede enunciar de la siguiente manera: Un programa de computadora aprende a partir de una experiencia *E* al realizar una tarea *T*, si su rendimiento al realizar *T*, medido con *P*, mejora gracias a la experiencia *E*.

Existen varias tareas que se pueden abordar con sistemas de aprendizaje. Éstas pueden clasificarse como:

1. *Tareas de predicción*: De manera general, estas tareas se pueden dividir en dos, categorización y regresión.

- La *categorización* es una tarea básica en el análisis de datos y en el reconocimiento de patrones. La categorización es la tarea en la cual un conjunto de objetos son ubicados dentro de un conjunto de categorías previamente definidas utilizando información etiquetada como entrenamiento. En este caso, un objeto puede pertenecer a una categoría, a más de una o a ninguna (Di Nunzio, 2009).

- La *regresión o estimación* tiene como objetivo inducir un modelo para poder predecir el valor futuro de una variable dados los valores presentes y pasados de los atributos. Por ejemplo, estimar la producción de gasolina en una refinería (Aurajo, 2006).

2. *Tareas descriptivas*: son usadas para el análisis preliminar de los datos. Buscan derivar descripciones concisas de características de los datos, por ejemplo, medias y desviaciones estándares, entre otras, que permitan describir a un conjunto de datos (Glenn et al., 2006).

3. *Tareas de segmentación*: tratan de buscar una separación de los datos en subgrupos o categorías de acuerdo a un cierto criterio. Las categorías pueden ser exhaustivas y mutuamente excluyentes o jerárquicas. En esta tarea es común utilizar algoritmos de clustering, principalmente cuando no se cuenta con un conjunto de entrenamiento (Patwardhan et al., 2007).

4. *Tareas de análisis de dependencias*: el valor de un elemento puede usarse para predecir el valor de otro. La dependencia puede ser probabilística, por medio de una red de dependencias o puede ser funcional (Aurajo, 2006).

5. *Tareas de detección de desviaciones de casos extremos o anomalías*: permiten detectar los cambios más significativos en los datos con respecto a valores pasados o normales y filtra grandes volúmenes de datos que son menos probables de ser seleccionados. El problema central de esta tarea se encuentra en determinar cuándo una desviación es significativa para ser de interés (Glenn et al., 2006).

6. *Tareas de aprendizaje en base a experiencia*: se utiliza información y retroalimentación de soluciones para mejorar el desempeño de un sistema basado en aprendizaje automático, entre más se utiliza el sistema, mayor será la experiencia adquirida y por tanto tendrá un mejor desempeño (Aurajo, 2006).

7. *Tareas de búsqueda:* se utilizan principalmente para resolver algún problema de optimización. Involucran el uso de algoritmos genéticos y técnicas de búsqueda local, con la finalidad de encontrar información relevante a una petición hecha por el usuario (Ferrández et al., 2008).

En base tanto a la definición de aprendizaje automático, como a las tareas que con él se pueden llevar a cabo utilizando sistemas de aprendizaje, podemos decir que ocurre aprendizaje automático en un programa, si este puede modificar aspectos de sí mismo, de tal modo que en una ejecución subsecuente con la misma entrada, se produce un resultado mejor. Uno de los tipos de aprendizaje automático es el aprendizaje supervisado donde al algoritmo de aprendizaje se le proporciona un conjunto de entrada con las correspondientes salidas correctas, y a partir de éstas el algoritmo "aprende" comparando su salida con la correcta, con esto sabe el error, luego entonces, se modifica para corregir (Aurajo, 2006). Un ejemplo de aplicación de este tipo de aprendizaje es la categorización automática de documentos.

Para la realización del presente trabajo se hace uso de algunas de estas tareas que utilizan el aprendizaje automático, en particular con las relacionadas con el aprendizaje semi-supervisado. En la siguiente sección se presenta la definición formal de esta tarea.

2.2 Aprendizaje semi-supervisado

El aprendizaje semi-supervisado surge como consecuencia de la dificultad que se tiene para obtener las instancias etiquetadas, requeridas por los métodos supervisados, ya que éstas deben ser etiquetadas por un experto en forma manual convirtiéndose en un trabajo tedioso, que consume mucho tiempo, además de requerir de un especialista del área en la cual se quiere llevar a cabo la categorización, con lo que, un cambio de área requiere generalmente un cambio de experto (Chapelle et al., 2006).

En el aprendizaje semi-supervisado se utiliza, generalmente, una colección limitada de instancias etiquetadas, las cuales son utilizadas por el sistema de categorización para a su vez "etiquetar" un conjunto de objetos no etiquetados. Las instancias no etiquetadas seleccionadas son incorporadas al conjunto de entrenamiento y por lo tanto utilizadas para seguir aprendiendo (Cohen et al., 2004).

En el caso de la categorización automática de documentos, los objetos son los documentos que se quiere categorizar. Los atributos de categorización son las palabras de estos documentos. Muchas veces estos atributos son tomados en función de cumplir con un umbral o condición para disminuir su número ya que entre más atributos se tengan se incrementa el costo computacional aunado a que no por tener más atributos se tendrán mejores resultados. Una vez que se han planteado las definiciones tanto de aprendizaje automático como de aprendizaje semi-supervisado, en el siguiente apartado se hace referencia a un tema que sirve como eje para el desarrollo del presente trabajo de tesis: la categorización automática de documentos.

2.3 Categorización automática de documentos

La categorización automática de documentos es la tarea en la cual los documentos, en formato electrónico, son categorizados dentro de un conjunto de categorías previamente definidas utilizando información etiquetada como entrenamiento. La categorización automática de documentos corresponde a una tarea descriptiva, como se comentó en el apartado anterior. Esta tarea tiene varias aplicaciones como son la categorización automática de correos electrónicos (Joachims, 1998) y la categorización de páginas Web (Aas et al., 1999). En las siguientes sub-secciones se presenta la descripción de los principales aspectos de la tarea de categorización automática de documentos.

2.3.1 Definición del problema

La *categorización* puede ser formalizada como la tarea de aproximar una *función objetivo* desconocida $\Phi : I \times C \to \{T, F\}$, la cual describe cómo las instancias del problema deben ser categorizadas de acuerdo a un experto en un dominio determinado como verdadero (T) o falso (F), por medio de una función $\Theta : I \times C \to \{T, F\}$ llamada el *categorizador*, donde $C = \{c_1, \ldots, c_{|c|}\}$ es un conjunto de categorías o categorías previamente definido, e I es un conjunto de instancias del problema.

Comúnmente cada instancia $i_j \in I$ es representada como una lista de valores característicos, también conocidos como atributos, representados por A, donde $A = \langle a_1, a_2, \ldots, a_{|A|} \rangle$. Por ejemplo, la representación de los atributos para la instancia i_j, estaría representado por $i_j = \langle a_{1j}, a_{2j}, \ldots, a_{|A|j} \rangle$, donde $|A|$ representa el número total de atributos con que cuenta la instancia i_j.

Si $\Phi : i_j \times c_i \rightarrow T$, entonces i_j es llamado un *ejemplo positivo* de c_i, mientras que sí $\Phi : i_j \times c_i \rightarrow F$, es llamado un *ejemplo negativo* de c_i. Para generar automáticamente el sistema de categorización de c_i es necesario un proceso inductivo, llamado *aprendizaje*, el cual por observación de los atributos de un conjunto de instancias previamente categorizadas bajo c_i o $\overline{c_i}$, adquiere los atributos que una instancia no vista debe tener para pertenecer a la categoría. Por tal motivo, en la construcción del sistema de categorización se requiere la disponibilidad inicial de una colección Ω de ejemplos tales que el valor de $\Phi(i_j, c_i)$ es conocido para cada $\langle i_j, c_i \rangle \in \Omega \times C$. A la colección usualmente se le llama *conjunto de entrenamiento*. En resumen, al proceso anterior se le identifica como *aprendizaje supervisado* debido a la dependencia del conjunto de entrenamiento.

Se pueden identificar con facilidad tres paradigmas en la categorización automática de documentos: el caso binario, el caso multi-categoría y el caso multi-etiqueta, los cuales se describen brevemente a continuación:

- En el caso binario una instancia a ser categorizada recibe exactamente una de dos posible etiquetas, tales como: sí o no, verdadero o falso, entre otras.

- En el caso multi-categoría una instancia a ser etiquetada recibe exactamente una categoría dentro de un conjunto (mayor a dos) de categorías posible.

- Finalmente, en el caso multi-etiqueta, una instancia puede recibir más de una categoría al llevar a cabo la tarea de categorización.

2.3.2 Representación de los documentos

Una vez obtenido el conjunto de documentos de entrenamiento, el siguiente paso en la construcción de un sistema de categorización de documentos consiste en transformar los documentos, de su formato inicial, a una representación adecuada para el algoritmo de aprendizaje, y en general para la tarea de categorización. Básicamente, esta parte del proceso considera la extracción de las características principales (conjunto de palabras) de los documentos de entrenamiento. A continuación se describen las diferentes etapas empleadas en la extracción de dichas características.

2.3.2.1 Pre-procesamiento

El primer paso en la categorización de documentos es la transformación de los documentos, los cuales típicamente son cadenas de caracteres, en una representación que sea compatible con algún algoritmo de aprendizaje utilizado para llevar a cabo la tarea de categorización de los documentos. La transformación textual usualmente consiste en las siguientes acciones:

- *Remover etiquetas*: Las etiquetas tanto en HTML como en XML, entre otras, se utilizan normalmente para organizar las colecciones tanto de entrenamiento como de prueba bajo diferentes categorías o rubros como puede ser temática, fecha o el nombre del autor. Sin embargo, éstas deben ser removidas para poder llevar a cabo la tarea de categorización utilizando sólo la información del texto en cuestión.

- *Remover palabras de paro*: Las palabras de paro, también llamadas *stopwords*, son palabras frecuentes (artículos, pronombres, preposiciones y conjunciones) que no transmiten información.

- *Uso de un lematizador*: Consiste en obtener las raíces morfológicas de las palabras. De esta manera, por ejemplo, la palabra caminar, camino, caminaré o caminé, serán llevados a una misma raíz léxica. Comúnmente se lleva a cabo esta tarea aplicando algún algoritmo de eliminación de afijos. Uno de los algoritmos más comúnmente utilizado para esta tarea es el sistema Porter Stemmer (Porter et al., 1980), que elimina sufijos de términos en inglés.

2.3.2.2 Indexado

El *indexado* denota la actividad de hacer el mapeo de un documento y representar en una forma compacta su contenido. La forma más común para representar cada documento es un vector con términos ponderados como entradas, concepto tomado del modelo de espacio vectorial usado en recuperación de información (Aas et al., 1999). Es decir, si la colección de documentos está representada por $D = \{d_1, d_2, ..., d_n\}$, donde d_i es un documento y n es el número de documentos en la colección, para cada documento se forma una representación matricial, por ejemplo el documento *k*-ésimo estará representado por $d_k = \{a_{1k}, a_{2k}, ..., a_{mk}\}$, donde *m* es el número de términos en la colección.

Existen varias maneras de determinar el peso w_{kj} del término *j* en el documento *k*, el peso se asocia comúnmente con la importancia que el término en cuestión tiene para distinguir una categoría de otra. Muchas de éstas se basan en las dos observaciones siguientes:

- Cuantas más veces aparece un término en un documento, más relevante es para dicho documento.

- Cuanto mayor es el número de documentos en los que aparece una palabra, menos nos sirve para discriminar entre los documentos.

Con respecto al peso w_{kj} se tienen diferentes formas de calcularlo. Entre las más usadas se tienen el ponderado Booleano, por frecuencia de término, el llamado $tf \cdot idf$ y el ponderado por frecuencia normalizada, los cuales se describen a continuación:

- *Ponderado Booleano*: Consiste en asignar el peso de 1 si el término ocurre en el documento y 0 en otro caso.

$$w_{kj} = \begin{cases} 1 \text{ si } t_k \text{ aparece en } d_j \\ 0 \text{ en otro caso} \end{cases}$$

- *Ponderado por frecuencia de término*: Consiste en asignar el número de veces que el término ocurre en el documento d_j.

$$w_{kj} = f_{kj}$$

- *Ponderado* $tf \cdot idf$: Combina la frecuencia del término en el documento con la frecuencia de éste en el resto de los documentos de la colección (Salton et al., 1987; Debole et al., 2003)

$$w_i = tf_i \cdot \log\left(\frac{N}{n_i}\right)$$

Donde N es el tamaño de la colección, es decir, el número total de documentos y n_i es el número de documentos en los que aparece el término *i*-ésimo.

- *Ponderado por frecuencia normalizada:* La combinación de la frecuencia en el documento con la ponderación *idf* suele ofrecer una mejora aún más importante.

Sin embargo, es muy importante normalizar de algún modo la frecuencia en el documento para moderar el efecto de los términos de alta frecuencia y compensar la longitud del documento, tal como se muestra en la siguiente ecuación (Debole et al., 2003):

$$w_{kj} = \frac{\log_2(f_{kj} + 1)}{\log_2 voc_j}$$

Donde f_{kj} es la frecuencia del término *k* en el documento *j*, mientras que voc_j es el numero de términos únicos en el documento *j*.

2.3.2.3 Reducción de dimensionalidad

Un problema central en la categorización automática de textos es la alta dimensionalidad del espacio de características o atributos. La dimensionalidad es dada por el número de palabras distintas, típicamente miles, que tiene una colección de documentos. Si se utilizan las técnicas estándar de categorización, cuando se tiene un conjunto de características muy grande, esto implica un alto costo computacional. Sin embargo, este costo, la mayoría de las veces, no representa una mejora significativa en los resultados obtenidos. De aquí surge entonces la necesidad de reducir la dimensionalidad del espacio de características.

En la categorización automática de documentos la alta dimensionalidad del espacio de términos puede ocasionar un *sobre-ajuste* en el proceso de aprendizaje (Aas et al., 1999), lo cual provoca problemas de efectividad debido a que el sistema de categorización tiende a comportarse mejor sobre los datos con los que ha sido entrenado y no conserva la tendencia en aquellos no vistos.

Además la alta dimensionalidad también se refleja en la eficiencia, haciendo el problema menos tratable para el método de aprendizaje. Para disminuir el problema se selecciona un subconjunto de *m* de atributos. Este proceso, que comúnmente se le conoce como *selección de características*, permite reducir significativamente la dimensionalidad, es decir, su efecto es reducir el tamaño del vector de características de m a m', siendo $m' << m$; donde el conjunto m' es llamado *conjunto de términos reducido*. La reducción se hace calculando una función de calidad para los términos, y seleccionando aquellos con mayor calificación.

En el presente trabajo, se empleó como función de reducción de dimensionalidad la ganancia de información (ver apartado 2.3.2.5), la cual se encuentra entre las más efectivas como se puede ver en diversas literaturas del estado del arte. Por ejemplo Yang (Yang et al., 1997) llevó a cabo experimentos, en los cuales se utilizaron diferentes técnicas de reducción de dimensionalidad y encontró que con esta función pueden ser eliminados hasta un 99% de términos del conjunto original, incrementando la exactitud del sistema de categorización y obteniendo una importante mejora en la eficiencia.

Cabe mencionar que existen otras medidas que permiten reducir la dimensionalidad del espacio de características, como son: umbral de frecuencia, χ^2, componentes principales y evaluación estilométrica, entre otras (Sebastiani, 2002). En los siguientes apartados se describe la reducción de dimensionalidad por umbral de frecuencia y por ganancia de información.

En la presente tesis se utiliza el umbral de frecuencia en la selección de características, en conjunto con la ganancia de información, la cual se describe en el siguiente apartado. La elección de estos métodos de reducción se debe a que evaluaciones presentadas en el estado del arte han revelado que estos métodos se encuentran entre los más efectivos (Yang et al., 1997; Montejo et al., 2007).

2.3.2.4 Umbral de frecuencia

El umbral de frecuencia elimina aquellos términos que no logran superar un umbral fijado previamente como punto de corte o de rechazo. Es decir, la idea básica es que los términos "raros" o poco frecuentes no proporcionan información para predecir la categoría y por lo tanto estarán por debajo del punto de corte, quedando fuera del conjunto seleccionado.

Al llevar a cabo un "recorte" del vocabulario al realizar la tarea de categorización se podría pensar que este hecho afectará la exactitud de categorización obtenida. Sin embargo, ocurre un fenómeno interesante, estudiado por George Kingsley Zipf y plasmado en la ley que lleva su nombre, Ley de Zipf. Esta ley afirma que un pequeño número de términos son utilizados con mucha frecuencia, mientras que frecuentemente ocurre que un gran número de términos son poco empleados (Yavuz, 1974). Esta afirmación, expresada matemáticamente quedaría de la siguiente forma:

$$F = \frac{k}{R}$$

Donde F representa la frecuencia de aparición de un término en una colección de documentos, y mantiene una relación inversa con su rango, R, y k es una constante. A manera de ilustrar la aplicación de la ley de Zipf, supongamos que el vocabulario mostrado en la tabla 2.1 corresponde a una colección referente a noticias de desastres naturales.

Si se grafica la relación resultante de la frecuencia F contra el rango R, se obtiene la gráfica que se muestra en la figura 2.1. Como se puede apreciar, en la medida en que la frecuencia de aparición disminuye las palabras se vuelve menos significativas, por ejemplo, para distinguir una categoría de otra. La Ley de Zipf se utiliza, además, para estudiar las propiedades estadísticas de grandes conjuntos de datos, como la Web (Gelbukh et al., 2001).

Tabla 2.1: Ejemplo de la ley de Zipf

R	F	Palabras
1	36	personas
2	25	estado
3	22	incendios
4	21	mil
5	19	lluvias
6	15	inundaciones
7	14	zona
8	13	kilómetros
9	13	incendio
10	11	afectadas
11	10	autoridades
12	8	daños
13	8	huracán
14	8	bomberos
15	7	sismo
16	7	nacional
17	6	centro
18	6	zonas
19	6	millones

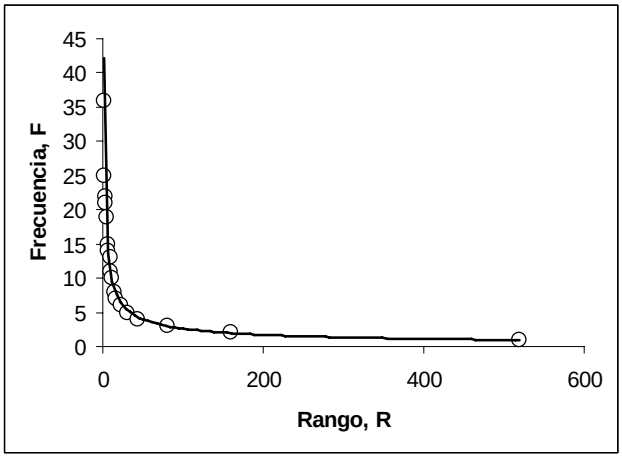

Figura 2.1: Ejemplo de la ley de Zipf

En la presente tesis utilizamos como umbral para la selección de términos más significativos las siguientes medidas: frecuencia y ganancia de información. La frecuencia de los términos seleccionados debe ser mayor que uno. El criterio considerado para elegir sólo los términos con frecuencia mayor que uno se basa en que una palabra refleja más el contenido de un documento entre más frecuente sea. En este orden de ideas, una palabra poco frecuente reflejaría poco el contenido del documento que la contiene.

Así, todas las palabras que aparecen en la colección una sola vez son eliminadas del conjunto de características, por considerar que su aporte para distinguir una categoría de otra es muy poco o nulo, lo que permite, además, tener un vector de características de menor dimensionalidad. La ganancia de información, la cual se detalla en el siguiente apartado, debe ser mayor que cero, debido a que estas son las palabras que tienen un mayor poder discriminativo. En el capítulo cuatro se retoma el uso de estas dos medidas en la selección de características.

2.3.2.5 Ganancia de información

Existen diversas técnicas para reducir el número de atributos, entre éstas se encuentra la ganancia de información, *Information Gain (IG)*, la cual es una medida basada en la entropía de un sistema (Lee et al., 2006), esto es, su grado de desorden (Shannon et al., 1963). IG indica cuánto se reduce la entropía de todo el sistema si se conoce el valor de un atributo determinado. Con esto se puede saber cómo este atributo está relacionado con respecto al sistema completo, en otras palabras, cuánta información aporta dicho atributo al sistema. Para un conjunto $C = \{c_1, c_2, ..., c_n\}$ de categorías posibles, la ganancia de información del atributo o término *t* se define como:

$$IG(t) = -\sum_{i=1}^{n} P(c_i)\log(P(c_i)) + P(t)\sum_{i=1}^{n} P(c_i|t)\log(P(c_i|t)) + P(\bar{t})\sum_{i=1}^{n} P(c_i|\bar{t})\log(P(c_i|\bar{t}))$$

Donde $P(c_i)$ es la probabilidad de la categoría c_i, $P(t)$ es la probabilidad de seleccionar un documento que contenga el término *t*, $P(c_i|t)$ es la probabilidad condicional de que un documento que contenga el término *t* pertenezca a la categoría c_i dado el documento con el término *t*; y por último, $P(c_i|\bar{t})$ es la probabilidad condicional de que un documento pertenezca a la categoría c_i dado que el documento no contiene el termino *t*.

Para la elección de los atributos se fija un umbral o punto de corte. El conjunto de atributos con ganancia de información menor que el umbral es eliminado.
En los experimentos realizados en el presente trabajo se utilizó la ganancia de información para la selección de los atributos. Al involucrar a todas las categorías se consideró a la ganancia de información como una medida global, como se detallará en el capítulo cuatro. Sólo fueron considerados los atributos con una ganancia de información positiva ($IG \geq 0$).

2.3.3 Algoritmos de categorización

Dentro del área de aprendizaje automático existen una gran variedad de algoritmos. Estos algoritmos se clasifican de acuerdo con su uso en diferentes métodos, ya sean estos supervisados, no supervisados o semi-supervisados, además del tipo de modelo de aprendizaje utilizado. Los modelos pueden ser probabilísticos, de agrupación, de co-entrenamiento y basados en grafos, entre otros (Di Nunzio, 2009).

Diferentes tipos de algoritmos han sido utilizados para llevar a cabo la tarea de categorización automática de documentos. La función principal de éstos, es realizar el proceso inductivo necesario, basado en el conjunto de entrenamiento, para asignar automáticamente una categoría, de entre varias previamente definidas, a un documento. Entre estos algoritmos se encuentran: árboles de decisión (Duda et al., 1973), vecinos más cercanos (Nigam et al., 2000), Rocchio (Riloff et al., 2005), el método naïve Bayes, NB (Peng et al., 2004) y las máquinas de vectores de soporte (Support Vector Machine, SVM) (Joachims, 1999), por mencionar algunos.

En este trabajo nos enfocamos en dos de los que se reportan resultados destacados en la tarea de categorización automática de documentos: naïve Bayes y las máquinas de vectores de soporte, estos métodos fueron utilizados en los experimentos realizados en la presente tesis y se describen a continuación.

2.3.3.1 Naïve Bayes

El método bayesiano o probabilístico ha sido ampliamente usado para la categorización de documentos (Lewis, 1998). Este método usa la probabilidad conjunta de las palabras y las categorías para estimar la probabilidad $P(c_i|d_j)$ de cada categoría, dado un documento.

Si se tiene un conjunto de documentos $D = \{d_1, d_2, ..., d_m\}$ asociado a las categorías previamente definidas $C = \{c_1, c_2, ..., c_n\}$, cada documento es representado por un vector $d_j = (w_{1j}, w_{2j}, ..., w_{|\tau|j})$ donde τ es el conjunto de términos que pertenecen a c_i; el método bayesiano estima la probabilidad a posteriori de cada categoría c_i dado el documento d_j, de la siguiente manera:

$$P(c_i \mid d_j) = \frac{P(c_i)P(d_j \mid c_i)}{P(d_j)}$$

Donde, $P(d_j)$ es la probabilidad de que se elija aleatoriamente el documento d_j (esta probabilidad es independiente de las categorías, por lo que se puede omitir) y $P(c_i)$ es la probabilidad de que el documento elegido pertenezca a la categoría c_i. Debido a que el número de posibles documentos d_j es muy grande, se vuelve complicado el cálculo de $P(d_j \mid c_i)$.

Para simplificar el cálculo de $P(d_j \mid c_i)$ es común asumir que la probabilidad de un término dado es independiente de los otros términos que aparecen en el mismo documento. Aunque a primera vista esto puede ser visto como una simplificación exagerada, el método naïve Bayes representa resultados comparables con los obtenidos por métodos más elaborados (Yang et al., 1999).

Usando esta simplificación es posible determinar $P(d_j \mid c_i)$ como el producto de probabilidades de cada término que aparece en el documento, de la siguiente manera:

$$P(d_j \mid c_i) = \prod_{t=1}^{\tau} P(w_{tj} \mid c_i)$$

De las dos expresiones anteriores, tenemos que la probabilidad de que el documento d_j elegido aleatoriamente pertenezca a la categoría c_i es:

$$P(c_i \mid d_j) = P(c_i) \prod_{t=1}^{|\tau|} P(w_{tj} \mid c_i)$$

Con $P(c_i)$ calculado como: $P(c_i) = \frac{N_{c_i}}{N}$; donde N_{c_i} es el número de documentos de la categoría c_i y N es el total de documentos en el conjunto de entrenamiento. Por su parte $P(w_{tj} \mid c_i)$ puede ser calculado como:

$$P(w_{tj} \mid c_i) = \frac{1 + count(w_{tj}, c_i)}{N_{c_i} + |\tau|}$$

Donde $count(w_{tj}, c_i)$ es el número de veces que el término w_{tj} aparece en los documentos de la categoría c_i. Para resolver el problema de probabilidad cero se usa la estimación de Laplace, conocida como: Add-One Smoothing (Jurafsky et al., 2000).

De esta manera a d_j se le asigna la categoría c_i, donde $P(c_i|d_j)$ es máxima. El método naïve Bayes es muy popular en el área de categorización de documentos, y ha sido utilizado por muchos investigadores entre los que destacan los siguientes (Di Nunzio, 2009; Eibe et al., 2006; Peng et al., 2004; Lewis et al., 1994; Koller et al., 1997).

2.3.3.2 Máquinas de vectores de soporte

Las máquinas de vectores de soporte fueron presentadas por Vapnik (Vapnik, 1995), y fueron aplicadas por primera vez a la categorización de textos por Joachims (Joachims, 1998; Burges, 1998). Esta técnica tiene raíces en la teoría de aprendizaje estadístico. Básicamente mapea los documentos en un espacio de atributos de alta dimensionalidad e intenta aprender hiperplanos de margen máximo entre dos categorías de documentos. Además, representa los límites de decisión usando un subconjunto de ejemplos de entrenamiento, conocidos como vectores de soporte.

La Figura 2.2 muestra una gráfica de un conjunto de ejemplos de entrenamiento que pertenecen a dos diferentes categorías representadas por cruces y círculos. Los datos son linealmente separables, es decir, podemos encontrar un hiperplano tal que todas las cruces estén de un lado del hiperplano y todos los círculos queden en el otro lado. Sin embargo, hay una infinidad de posibles hiperplanos, como se puede apreciar en la figura. El hecho de que estos hiperplanos no tengan ningún error al separar los ejemplos de entrenamiento, no garantiza que con nuevos documentos suceda lo mismo.

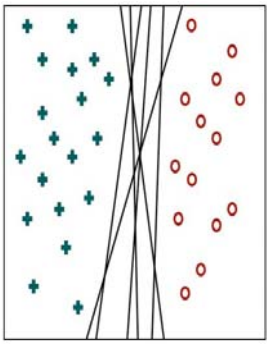

Figura 2.2: Problema de categorización linealmente separable.

La Figura 2.3, muestra dos hiperplanos y sus márgenes de riesgo de error. Entre mayores son los márgenes, menor será el riesgo de que un documento nuevo sea categorizado de manera errónea. En esta figura los cuadros indican los ejemplos que son tomados como vectores de soporte. Esto es para el caso en que los conjuntos son linealmente separables.

Para conjuntos de documentos que no son linealmente separables, SVM usa *funciones de convolución* o *Kernels*. Estos Kernels transforman el espacio de atributos iniciales en otro, donde los documentos transformados son linealmente separables. Un método basado en SVM puede encontrar el hiperplano que separa los documentos con el valor máximo. Para una descripción detallada de SVM ver (Joachims, 2002).

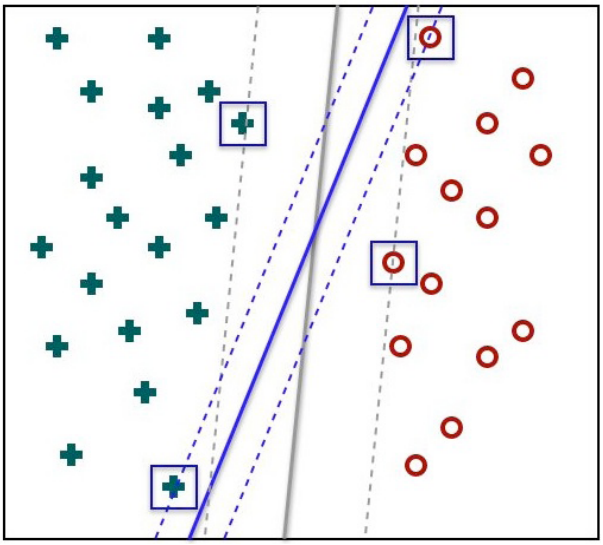

Figura 2.3: Un par de hiperplanos y sus márgenes de riesgo de error.

Esta idea puede ser generalizada fácilmente para colecciones con más de dos categorías de documentos, la idea es dividir el problema multi-categoría y convertirlo en varios problemas binarios. Generalmente es usado en uno contra uno o uno contra todos (Scholkopf et al., 2003). Para uno contra uno, si se tienen q categorías, se construyen $q(q-1)/2$ sistemas de categorización usando los documentos de cada combinación de dos categorías distintas. Para determinar la categoría del documento nuevo se usa una estrategia de voto. En uno contra todos, se construyen q sistemas de categorización para cada categoría, usando los ejemplos de una categoría y mezclando todas las demás categorías. En este segundo caso, el sistema de categorización produce una función que da un valor relativamente mayor a una de las dos categorías, al documento nuevo se le asigna la categoría que obtuvo el valor más alto.

En uno contra uno se construyen más sistema de categorización, pero cada uno de ellos tiene menos ejemplos de entrenamiento. La categorización uno contra uno ha mostrado ser mejor en la práctica (Joachims, 2001). Se han hecho amplias comparaciones reportando que de los sistemas de categorización disponibles actualmente, el método SVM es el que mejores resultados obtiene en la mayoría de los casos (Di Nunzio, 2009; Joachims, 1998; Yang et al., 1999; Liu et al., 2005; Cardoso et al., 2006).

2.4 Evaluación de un sistema de categorización

La evaluación de los sistemas de categorización automática de documentos generalmente se lleva a cabo de manera experimental y rara vez se realiza de manera analítica. La razón es porque, con la finalidad de evaluar analíticamente un sistema se requieren especificaciones formales del problema que el sistema está tratando de resolver.

La evaluación experimental de un sistema de categorización usualmente mide su exactitud, esto es, su habilidad para tomar las decisiones correctas de categorización. En los siguientes apartados se presenta una descripción de las medidas de evaluación más utilizadas en sistemas de categorización automática de documentos.

2.4.1 Medidas de evaluación

La evaluación del desempeño de un sistema de categorización es un factor muy importante del aprendizaje computacional. La forma más común de medir su eficiencia es mediante la exactitud predictiva. Cada vez que se introducen nuevos ejemplos a un sistema de categorización, éste debe tomar la decisión correcta sobre la etiqueta que le asignará. La categorización incorrecta de un ejemplo se considera como un error del sistema de categorización. La tasa de error, o su complementaria, la tasa de acierto, se calcula de la siguiente manera (Aurajo, 2006):

$$\text{tasa de error} = \frac{\text{Número de errores}}{\text{Número total de casos}}$$

Además de este concepto, también existen otras medidas de evaluación del desempeño de un sistema de categorización, las cuales se describen a continuación.

Para categorizar colecciones en las que se tienen únicamente dos categorías se tienen las siguientes medidas:

- Verdaderos Positivos (*TP*, por sus siglas en inglés): Son aquellas instancias cuya hipótesis dice que deben ser positivos y en realidad son positivos.

- Verdaderos Negativos (*TN*, por sus siglas en inglés): Son aquellas instancias cuya hipótesis dice que deben ser negativos y en realidad son negativos.

- Falso Positivo (*FP*): Son aquellas instancias cuya hipótesis dice que deben ser positivos y en realidad son negativos.

- Falso Negativo (*FN*): Son aquellas instancias cuya hipótesis dice que deben ser negativos y en realidad son positivos.

A partir de las medidas anteriores surgen otras medidas de evaluación de un sistema de categorización las cuales se presentan a continuación:

- *Precisión*: Porcentaje o proporción de predicciones positivas que son correctas, es decir, es la probabilidad de que una instancia *x* sea categorizada con la categoría *c*, y esta instancia realmente pertenezca a esta categoría.

$$\text{Precisión} = \frac{TP}{TP + FP}$$

- *Cobertura*: Porcentaje de verdaderos positivos predichos de entre todos los positivos. Es decir, es la probabilidad de que si una instancia *x* pertenece a una categoría *c*, el sistema de categorización le asigne la categoría correcta. A esta medida también se le conoce como *Recuerdo*.

$$\text{Cobertura} = \frac{TP}{TP + FN}$$

- *Exactitud*: Porcentaje de predicciones que les es asignada la categoría correctamente.

$$\text{Exactitud} = \frac{TP + TN}{TP + TN + FP + FN}$$

- *Especificidad*: Porcentaje de instancias negativas que fueron predichas como negativas.

$$\text{Especificidad} = \frac{TN}{TN + FP}$$

Las medidas anteriores pueden ser calculadas sobre una colección completa y en este caso reciben el nombre de micro-promedio, o bien, para cada categoría y enseguida promediar entre las categorías, lo que se conoce como macro-promedio.

En el *micro-promedio*, cada documento vale lo mismo para el promedio y las categorías pequeñas tienen poco impacto en el resultado final. En este caso la precisión y la cobertura se calculan de acuerdo con las siguientes expresiones, donde las c_i son las categorías:

$$\text{Precisión} = \frac{\sum_{i=1}^{c} TP_i}{\sum_{i=1}^{c} TP_i + FP_i}$$

$$\text{Cobertura} = \frac{\sum_{i=1}^{c} TP_i}{\sum_{i=1}^{c} TP_i + FN_i}$$

En el *macro-promedio*, primero se determina el promedio para cada categoría y considerando que cada categoría vale lo mismo para el promedio final. En este caso la precisión y la cobertura se calculan de acuerdo con las siguientes expresiones:

$$\text{Precisión} = \frac{\frac{\sum_{i=1}^{c} TP_i}{\sum_{i=1}^{c} TP_i + FP_i}}{c}$$

$$\text{Cobertura} = \frac{\frac{\sum_{i=1}^{c} TP_i}{\sum_{i=1}^{c} TP_i + FN_i}}{c}$$

Estas medidas son particularmente importantes cuando la colección es desbalanceada, esto es, cuando hay categorías con una gran diferencia en el número de documentos proporcionados como instancias de entrenamiento.

2.4.2 Estrategias de evaluación

Una buena medida de estimación del comportamiento del sistema de categorización con nuevos datos desconocidos es la exactitud. Sin embargo, si se calcula la precisión con el conjunto de ejemplos que se utilizó para entrenar el sistema de categorización, se obtiene con frecuencia una precisión mayor a la real, es decir, serán estimaciones muy optimistas al utilizar los mismos ejemplos en la inducción del algoritmo y en su comprobación (Sánchez, 2005).

La idea básica es estimar el desempeño de un sistema de categorización con una porción de los ejemplos y luego comprobar su validez con el resto de los ejemplos. Esta separación es necesaria para garantizar la independencia de la medida de precisión resultante, de no ser así, la exactitud del modelo será sobrestimada. Para tener seguridad de que las predicciones sean robustas y precisas, se consideran dos etapas en el proceso de construcción de un sistema de categorización, las cuales son: entrenamiento y prueba, partiendo los datos en dos conjuntos, un conjunto será de entrenamiento y otro de prueba. La finalidad es que los elementos del conjunto de prueba no sean vistos por el conjunto de entrenamiento.

Existen varias estrategias de validación de un sistema de aprendizaje dependiendo de cómo se realice la partición del conjunto de datos. Se utilizan dos técnicas para validar un sistema de categorización: la validación simple y la validación cruzada. El método de evaluación más básico, la validación simple, se describe en el siguiente apartado.

2.4.2.1 Validación simple

El método de evaluación más básico es el de validación simple. Este método utiliza un conjunto de muestras para construir el modelo del sistema de categorización, y otro diferente para estimar el error, con el fin de eliminar el efecto de la sobre-especialización (Sánchez, 2005).

De entre la variedad de porcentajes utilizados para formar estos conjuntos se tiene, entre los más frecuentes, el tomar 2/3 partes de las muestras para el proceso de aprendizaje y el 1/3 restante para comprobar el error del sistema de categorización. A estos conjuntos comúnmente se llama conjunto de entrenamiento y conjunto de prueba respectivamente. Cabe mencionar que estas fracciones (2/3 y 1/3) corresponden al número de instancias requeridas por los sistemas de categorización de documentos tradicionales, los cuales requieren, un mayor número de instancias de entrenamiento. El hecho de que sólo se utiliza una parte de las muestras disponibles para llevar a cabo el aprendizaje, es el inconveniente principal de esta técnica, al considerar que se pierde información potencialmente útil en el proceso de inducción del sistema de categorización. Sin embargo, a pesar de esto, éste es el escenario de validación deseable a la hora de evaluar el desempeño de un sistema de categorización automática de documentos, es decir, contar con un conjunto de instancias para llevar a cabo la evaluación que no hayan sido vistas por el conjunto de entrenamiento.

2.4.2.2 Validación cruzada

La validación cruzada es un método efectivo para probar la validez de los algoritmos y evitar pérdida de información (Kohavi, 1995). En esta técnica el conjunto de datos *D* se divide aleatoriamente en *k* particiones mutuamente excluyentes $D_1, D_2, \ldots D_k$, conteniendo cada una el mismo número de ejemplos, aproximadamente. La validación cruzada se ejecuta *k*-veces, por ello esta técnica es llamada en muchos casos como validación cruzada con *k* particiones o *k-fold-cross validation.*

Un valor comúnmente usado para *k* es 10. En cada evaluación se utiliza uno de los subconjuntos como conjunto de prueba, y se entrena el sistema con los *k*-1 conjuntos restantes. Así, la precisión estimada de categorización es la media de las *k* tasas obtenidas. La ventaja de usar *k-fold-cross validation* es que todos los ejemplos en el conjunto de datos son eventualmente usados para entrenamiento y prueba. En este caso no es necesario contar con un conjunto de prueba. Este hecho permite aprovechar al máximo la información proporcionada por las instancias de entrenamiento. La estimación de la exactitud de *k-fold-cross validation* es el número completo de categorizaciones correctas sobre el número de instancias en el conjunto de datos. Este esquema de validación es comúnmente usado cuando el numero de instancias con las que se cuenta no son suficientes como para formar los conjuntos de entrenamiento y prueba.

2.5 Conclusiones del capítulo

En este capítulo se hace una introducción a la teoría que fundamenta la presente tesis. Los conceptos que se revisan van desde el aprendizaje automático y la importancia que éste tiene en la solución de problemas de categorización de documentos. Específicamente se aborda desde la representación que se debe hacer con los documentos, así como la parte del pre-procesamiento y la importancia que tiene el poner los documentos en una misma representación, para que sea entendible por una máquina. También se aborda la reducción de la dimensión del vector de características y las distintas maneras de llevar a cabo esta tarea con la finalidad de que el costo computacional sea menor sin sacrificar la exactitud de categorización. Específicamente se revisan las técnicas de reducción de dimensionalidad basadas en umbral de frecuencia y en ganancia de información, medidas que hemos empleado en la selección de los términos en los experimentos realizados. El empleo de estas medidas permite tener las mejores características a nivel local y global respectivamente, garantizando con esto la selección de los términos más representativos para una categoría así como los de mayor peso a nivel global.

Se presentan, además, los algoritmos de categorización que hemos utilizado en los experimentos realizados en el presente trabajo: naïve Bayes y SVM, así como la manera en que es evaluada la tarea de categorización automática resaltando las medidas de exactitud, precisión y cobertura las cuales nos dan una clara idea del desempeño del sistema de categorización bajo estudio. Cabe mencionar que estas medidas de evaluación son utilizadas de manera conjunta con dos estrategias de evaluación: validación simple y validación cruzada. El uso de estas estrategias está en función de contar o no con un conjunto de prueba independiente del conjunto de entrenamiento.

Capítulo III

Trabajo relacionado

En este capítulo se presenta una descripción de las principales técnicas y métodos desarrollados para resolver distintas problemáticas relacionadas con la tarea de categorización automática de documentos. Al ser estas problemáticas tan variadas y cada una de ellas tan amplia, no se pretende llevar a cabo una descripción exhaustiva de las mismas. La finalidad es proporcionar al lector un marco teórico que permita entender el desarrollo del presente trabajo de investigación tanto en su operación así como en su evaluación.

3.1 Introducción

En los últimos años, las técnicas de recuperación de información han adquirido gran popularidad debido a su eficacia para resolver problemas de distinta índole como el descubrimiento de medicamentos, diagnósticos médicos, predicción del clima, detección de fraudes, reconocimiento de caracteres, detección de anormalidades en los cromosomas y recuperación de imágenes, entre otros. En particular, la categorización automática de documentos ha sido estudiada extensamente en estadística, aprendizaje computacional, redes neuronales y sistemas expertos (Mitchell, 1997; Aurajo, 2006; Di Nunzio, 2009).

En el desarrollo de la presente tesis se abordan diferentes problemáticas dentro de la categorización automática de documentos, tanto temática como no temática: la categorización de documentos usando muy pocos ejemplos de entrenamiento, la categorización de documentos con un alto grado de traslape entre las categorías, la atribución de autoría y la desambiguación del sentido de las palabras. Todas estas tareas tienen un problema en común: la dificultad de contar con suficientes instancias manualmente etiquetadas que puedan ser utilizadas como entrenamiento.

En este capítulo se presenta una descripción de los principales trabajos relacionados con la presente tesis. En particular se abordan los siguientes temas: ensambles de sistemas de categorización, categorización semi-supervisada, uso de la Web como corpus, categorización con clases desbalanceadas y categorización no temática. Todos estos temas tienen relación con el desarrollo del método propuesto en el presente trabajo de tesis. En la siguiente sección se presenta una descripción de los principales ensambles utilizados en el estado del arte para la tarea de categorización automática de documentos.

3.2 Ensambles de clasificadores

Encontrar un sistema de categorización eficiente no es una tarea fácil. Cada sistema de categorización se caracteriza porque emplea una representación diferente de los datos. Encontrar una buena representación de estos requiere de tiempo y de varios ensayos previos. Aún no existe una regla general acerca de qué métodos de categorización son los más apropiados para qué tipos de problemas. Estudios previos han demostrado que ciertas partes del espacio de datos son mejor modeladas por un método de categorización en comparación con otros. El empleo de diferentes sistemas de categorización puede proporcionar información complementaria importante sobre la representación de los datos. Esto ha originado que utilizar una combinación o ensamble de sistemas de categorización sea una buena alternativa (Dzeroski et al., 2000).

Un ensamble de sistemas de categorización es un conjunto de métodos de categorización cuyas decisiones individuales para categorizar nuevos ejemplos son combinadas mediante algún esquema de toma de decisiones, normalmente por votación (Bennett et al., 2005). Los ensambles obtienen mayor precisión de categorización que la obtenida por sus componentes de manera individual.

Existen varias razones que justifican el ensamble de sistemas de categorización. Algunas de estas son:

(i) Los datos para entrenamiento pueden no proveer suficiente información para elegir un único sistema de categorización debido a que el tamaño disponible en estos datos es pequeño en comparación al espacio de hipótesis (Dietterich, 2000);

(ii) La combinación redundante y complementaria de sistemas de categorización mejora la robustez, precisión y generalidad de toda la categorización (Kotsiantis et al., 2004);

(iii) Diferentes sistemas de categorización utilizan diferentes técnicas y métodos de representación de los datos, lo que permite obtener resultados de categorización con diferentes patrones de categorización (Bennett et al., 2005);

(iv) Los ensambles son frecuentemente mas exactos que los sistemas de categorización individuales (Dzeroski et al., 2000).

A continuación se describen algunos de los ensambles más utilizados en tareas de categorización automática de documentos.

3.2.1 Bagging

El ensamble de clasificadores *bagging* (*bootstrap aggregating*), fue creado por Breiman en 1996 (Breiman, 1996). En este método, cada sistema de categorización se construye usando un conjunto de datos diferente, que es una muestra con reemplazamiento del conjunto de datos original. Normalmente el tamaño de la muestra coincide con el del conjunto de datos de partida, pero los conjuntos de datos difieren ya que en una muestra una instancia puede seleccionarse varias veces mientras que otra puede no seleccionarse (Eibl et al., 2005). Este ensamble consiste en agregar, mediante votación simple, los resultados de varios sistemas de categorización obtenidos de un mismo conjunto de entrenamiento mediante *muestreo con reemplazamiento* o *bootstrap* (Aurajo, 2006).

Una muestra de ejemplos bootstrap se genera al muestrear uniformemente *m* instancias del conjunto de entrenamiento con reemplazo. Se generan T muestras, $B_1, \ldots, B_T$ y se construye un sistema de categorización C_i para cada muestra. Con estas muestras se construye un sistema de categorización final C^* de todos los C_1 a C_T cuyo resultado es la salida mayoritaria de los sistemas de categorización (Breiman, 1994).

Para crear el primer sistema de categorización C_i se construye una muestra de igual tamaño que la original, pero obtenida mediante extracción con reemplazo. Para una de las muestras, un ejemplo tiene la probabilidad de $1-(1-1/m)^m$ de ser seleccionado por lo menos una vez en las m veces que se selecciona una instancia. Para valores grandes de m esto se aproxima a $1 - 1/e$=63.2 %. Por lo que cada muestra tiene aproximadamente un 63% de probabilidad de aparecer en los ejemplos de entrenamiento (Quinlan, 1996).

3.2.2 Stacking

Uno de los métodos existentes para la construcción de ensambles es el método *stacked generalization*, conocido también como stacking. Este método combina múltiples modelos que han sido entrenados para una tarea de categorización, es decir, combina varios sistemas de categorización para inducir un sistema de categorización de nivel más alto y con mayor rendimiento (Ting et al., 1997; Wolpert 1992).

Un ensamble de sistema de categorización tipo stacking construye un conjunto de modelos usando diferentes algoritmos de aprendizaje. Una forma de combinar los sistemas de categorización es usando voto mayoritario, sin embargo, esto tiene sentido cuando los sistema de categorización se desempeñan en forma parecida. Para producir una categorización se utiliza un meta-algoritmo (meta learner) que aprende de acuerdo a las salidas de los sistema de categorización base (en lugar de voto mayoritario). Esto es, se construyen N sistema de categorización a partir de los datos usando algoritmos diferentes (Mihalcea, 2004). La salida del stacking es el resultado de la votación de cada algoritmo de aprendizaje. Esta idea básica puede ser aplicada en muchas variantes, en función de los algoritmos utilizados y condiciones de cada uno de ellos, por ejemplo utilizando bagging y boosting (véase apartado siguiente). En el diseño de un arreglo stacking normalmente se utilizan modelos de diferente tipo, por ejemplo, el uso de SVM con naïve Bayes (Wolpert, 1992)

El procedimiento de votación de los diferentes algoritmos comúnmente es reemplazado por el concepto de meta-aprendizaje. El problema con el sistema de votación es que no queda claro cuál sistema de categorización dice la verdad. El arreglo stacking intenta mejorar la salida, usando otro algoritmo de aprendizaje, para descubrir cuál es la mejor opción de combinación que proporcione la salida más confiable en base al meta-aprendizaje. Las salidas de los sistemas de categorización se usan como atributos (por lo que se tienen tantos atributos para el meta-clasificador como sistema de categorización) de un nuevo sistema de categorización (Wolpert, 1992). Como meta-clasificador normalmente se utiliza un arreglo de sistemas de categorización conocidos, de los más comúnmente utilizados, como son naïve Bayes y SVM. En la tarea de categorización automática de documentos, se han hecho diversas aproximaciones combinando varios sistemas de categorización con la finalidad de mejorar la exactitud de categorización.

3.2.3 Boosting

Los métodos basados en *boosting* son algoritmos que en el enfoque supervisado conocen la solución *a priori* y utilizan esta información para adaptar su comportamiento. El *boosting* se basa en la pregunta *¿Puede un conjunto de sistema de categorización débiles crear un solo clasificador fuerte?* Un sistema de categorización débil es definido como un sistema de categorización que se correlaciona levemente con la categorización final o verdadera. En cambio, un sistema de categorización fuerte es un sistema de categorización que se correlaciona fuertemente con la categorización verdadera.

Después de que se agrega un sistema de categorización débil, los datos se vuelven a pesar. Los ejemplos que son categorizados correctamente pierden peso y los que son categorizados de manera errónea aumentan su peso. Así, los siguientes sistemas de categorización se centran más en los ejemplos que los anteriores categorizaron erróneamente.

El algoritmo boosting fue propuesto por Freund en 1996 y una de sus variantes más conocidas, es el algoritmo adaboost (Freund et al., 1996) el cual se describe también en este apartado. Al igual que bagging, el algoritmo boosting genera un conjunto de sistemas de categorización y luego realiza una votación entre ellos. Sin embargo, existe una diferencia sustancial entre los dos algoritmos debido a que bagging genera los sistemas de categorización de manera independiente, mientras que boosting no, ya que en éste los sistemas de categorización se generan secuencialmente.

En el algoritmo boosting a todos los ejemplos se les asigna inicialmente un peso igual ($1/m$), donde m es el número de instancias. Cada vez que se genera un sistema de categorización, los pesos de los ejemplos de la muestra de entrenamiento se actualizan de acuerdo a los resultados hallados previamente por el sistema de categorización, con el objetivo de minimizar el error esperado en las diferentes distribuciones de salida (Chapelle et al., 2006).

Dado el número de muestras bootstrap T, se generan secuencialmente T muestras de entrenamiento ponderadas y se construyen T sistemas de categorización $C_1, C_2, ..., C_T$. Finalmente se forma el sistema de categorización C_i, usando el esquema de votación ponderada de cada sistema de categorización.

El peso de cada sistema de categorización depende de su rendimiento en el conjunto de entrenamiento que se uso para construirlo (Schapire et al., 2000). Adaboost, por otro lado, es un método de aprendizaje iterativo que combina una serie de sistemas de categorización base usando una combinación lineal pesada para categorizar nuevos ejemplos. En cada iteración se genera un nuevo sistema de categorización, el cual trata de minimizar el error esperado asignando más peso a los ejemplos que fueron mal categorizados, aumentando así la probabilidad de que sean categorizados correctamente.

Formalmente, el algoritmo adaboost parte de un conjunto S de instancias etiquetadas donde a cada instancia x_i se le asigna un peso, $w(x_i)$. Se consideran N categorías, donde la categoría conocida de cada instancia x_i está dada por y_i. El sistema de categorización base es C, y h_t es la hipótesis parcial de cada uno de sistemas de categorización base del ensamble en cada iteración t, donde $t \in \{1,2,\ldots,T\}$. El peso es inversamente proporcional al error de cada sistema de categorización sobre los datos de entrenamiento (Freund et al., 1996).

Se han implementado diversas aproximaciones que utilizan boosting para llevar a cabo tareas de categorización automática de documentos (Schapire et al., 2000; Weiss et al., 1999). Mucho del trabajo previo que lleva a cabo la combinación de sistemas de categorización ha sido centrado en el uso de políticas básicas para la selección del mejor sistema de categorización o para combinar la salida de sistemas de categorización múltiples.

A manera de ejemplo se menciona el trabajo desarrollado por Schapire (Schapire et al., 2000), el cual utiliza un pesado para determinar la mejor salida basado en la combinación lineal de las respuestas parciales de los sistemas de categorización que componen el arreglo; Freund (Freund et al., 1996) utiliza combinaciones lineales de las probabilidades parciales asignadas por cada uno de los sistemas de categorización que componen el arreglo, mientras Yang (Yang et al., 1999), utiliza combinaciones lineales de las salidas parciales normalizadas.

También existe una gran variedad de trabajos que utilizan boosting para llevar a cabo tareas de categorización complementadas con aprendizaje semi-supervisado, entre estos trabajos destacan los siguientes (Chapelle et al., 2005; Jin et al., 2007; Eibl et al., 2005; Li, 2006). Dentro de los algoritmos propuestos que utilizan boosting y aprendizaje semi-supervisado para llevar a cabo la tarea de categorización destacan los siguientes (d´Alche et al., 2002; Bennett et al., 2005; Chen et al., 2007). Como se puede apreciar el común denominador en estos casos es el buscar algún tipo de estrategia que permita incrementar la exactitud de categorización, ya sea ésta probabilística o por medio de combinaciones lineales de probabilidades.

3.3 Categorización semi-supervisada

Existen diferentes aproximaciones para llevar a cabo la tarea de categorización automática de documentos. Específicamente existe la aproximación supervisada, no supervisada y semi-supervisada. De éstas, la diferencia entre las aproximaciones supervisada y no supervisada es la necesidad de un conjunto grande de instancias de entrenamiento manualmente etiquetado para la aproximación supervisada, los cuales muchas veces son difíciles de obtener. Sin embargo, esta aproximación es la que mejores valores de exactitud ofrece (Chapelle et al., 2005).

La categorización semi-supervisada es una forma especial de categorización en donde se usa un pequeño conjunto de datos manualmente etiquetados como entrenamiento y una gran cantidad de datos no-etiquetados, algunos de los cuales son incorporados al conjunto de entrenamiento con la finalidad de maximizar el nivel de predicción de categorías (Zhu, 2005; Tang et al., 2007; Mallapragada, et al., 2007). Este enfoque surge a raíz de que las instancias etiquetadas son frecuentemente difíciles de obtener para algunas tareas de categorización (por ejemplo, en atribución de autoría) aunado a que el proceso de etiquetado suele ser costoso y tedioso. De hecho, ésta es una de las principales desventajas de los método supervisados. En cambio los datos no etiquetados se pueden obtener de manera más simple, por ejemplo de la Web.

Un típico método de aprendizaje semi-supervisado en general consta de los cuatro pasos siguientes (Chapelle, 2006):

1. Entrenar un sistema de categorización usando un conjunto E de ejemplos de entrenamiento manualmente etiquetados.

2. Etiquetar ejemplos de entrenamiento no-etiquetados y obtener sus probabilidades de predicción usando E como entrenamiento. Seleccionar un subconjunto E' de instancias no etiquetadas en base a los criterios de selección establecidos previamente.

3. Entrenar el sistema de categorización con el nuevo conjunto formado por la unión de los conjuntos $E\, y\, E'$, esto es: $E \cup E'$.

4. Predecir las etiquetas del conjunto de prueba al utilizar $E \cup E'$ como entrenamiento.

Durante varias décadas, los estadísticos se han enfocado a utilizar una combinación de datos tanto etiquetados como no etiquetados para entrenar sistemas de categorización a través de modelos iterativo-incrementales. La idea de que un sistema de categorización aprenda utilizando una combinación de datos etiquetados y no etiquetados no es nuevo. Al menos desde 1968 se sugirió que se podían utilizar datos etiquetados y no etiquetados para construir sistemas de categorización (Hartley, 1968). Más recientemente, en el campo del aprendizaje automático, el uso combinado de ejemplos etiquetados y no etiquetados se ha aplicado de manera satisfactoria en varias tareas del procesamiento del lenguaje natural (Blum et al., 1998).

Una de las principales ventajas de los métodos semi-supervisados es que utilizan información no etiquetada, la cual es incorporada al conjunto de entrenamiento. Comúnmente se presenta esta información como una bolsa de instancias de la misma área o tópico en el mismo formato que las instancias etiquetadas. La finalidad de incorporar esta información al conjunto de entrenamiento es la de mejorar la exactitud obtenida incorporando información que nos permita incrementar la diferencia entre las categorías.

Una descripción detallada de métodos de categorización automática que utilizan métodos semi-supervisados se encuentra en las siguientes referencias (Chapelle et al., 2006; Chawla et al., 2004). Dentro de los algoritmos que utilizan el aprendizaje semi-supervisado se encuentran: self-training y co-training los cuales se explican a detalle en los siguientes apartados.

3.3.1 Self-training

Self-training es un proceso de aprendizaje semi-supervisado, donde se inicia con un conjunto de datos etiquetados para crear un sistema de categorización con el que posteriormente se etiquetan datos no etiquetados. Con este nuevo conjunto se crear un nuevo sistema de categorización y así hasta que se cumpla alguna condición. El funcionamiento de self-training es el siguiente:

(i) Primero se entrena a un sistema de categorización con una cantidad pequeña de ejemplos manualmente etiquetados.
(ii) Después, el modelo generado se utiliza para categorizar ejemplos no etiquetados.
(iii) De esta categorización se incorporan al conjunto de entrenamiento sólo aquellos ejemplos no etiquetados en los cuales su etiqueta predicha es más confiable.
(iv) El sistema de categorización se vuelve a entrenar y el proceso se repite

De esta manera el sistema de categorización utiliza sus mismas predicciones para aprender el mismo (Zhu, 2005). Self-training ha sido utilizado en varias tareas de procesamiento de lenguaje natural. Algunas de estas aplicaciones están enfocadas a resolver distintos problemas de categorización de documentos, como son la categorización con muy pocos ejemplos de entrenamiento (Brank et al., 2003), la categorización con categorías desbalanceadas (Chawla et al., 2004; Eibe et al., 21006), la atribución de autoría (Argamon et al., 2005;) y la desambiguación del sentido de las palabras (Navagli, 2009). Existen muchos algoritmos semi-supervisados en el estado del arte. Dentro de estos se encuentran: los métodos semi-supervisados basados en densidad (Bennett et al., 1999; Chapelle et al., 2005); algoritmos basados en grafos (Blum et al., 2001; Zhu, 2005; Belkin et al., 2004) y técnicas de boosting (Bennett et al., 2005; Chen et al., 2007). La mayoría de estos métodos fueron originalmente diseñados para problemas de dos categorías, sin embargo, muchas aplicaciones del mundo real requieren una categorización multi-categoría. En el presente trabajo de investigación se realizaron pruebas experimentales precisamente para estos problemas, los resultados experimentales obtenidos se presentan en los capítulos cinco y seis de esta tesis.

3.3.2 Co-training

El algoritmo co-training es un ensamble propuesto por Blum y Michell en 1998 (Blum et al., 1998). Este algoritmo asume que existen dos conjuntos de atributos independientes y compatibles para los ejemplos. Cada conjunto de atributos es suficientemente independiente para propósitos de aprendizaje y categorización. Un sistema de categorización que aprende de cada uno de estos conjuntos de atributos se puede usar para etiquetar datos para otro sistema de categorización y así expandir el conjunto de entrenamiento de ambos sistemas de categorización.

La idea principal del algoritmo co-training es que si dos algoritmos usan diferentes representaciones de sus hipótesis entonces pueden aprender dos modelos que pueden complementarse mutuamente. En algún momento uno de los algoritmos etiquetará algunos ejemplos no etiquetados y aumentará el conjunto de entrenamiento del otro algoritmo. Por ejemplo, si A y B son dos sistemas de categorización, A etiqueta datos para B y B etiqueta datos para A hasta que ya no existan datos no etiquetados o hasta que ningún dato pueda ser etiquetado debido a la incertidumbre en la asignación de la etiqueta a esos datos. La decisión para etiquetar datos del otro sistema de categorización se toma en base a técnicas estadísticas.

El conjunto de entrenamiento formado con los ejemplos etiquetados por ambos sistemas de categorización permite al primer sistema de categorización aprovechar la información acerca de la función objetivo con los ejemplos etiquetados por el segundo sistema de categorización.

Dentro de los trabajos encontrados en el estado del arte que recurren a este tipo de métodos y técnicas para llevar a cabo la tarea de categorización automática de documentos se encuentran el método de Zelikovitz (Zelikovitz et al., 2006) quienes usaron datos no etiquetados para incrementar la precisión de categorización utilizando como algoritmo de aprendizaje vecinos más cercanos (k-NN).

Su propuesta se basa en buscar una correspondencia entre las instancias de prueba directamente a "su ejemplo más cercano" del conjunto de entrenamiento y en base a ello asigna la nueva etiqueta. Como se mencionó anteriormente, el método propuesto está basado en la Web, por esta razón en el apartado siguiente se muestra el uso de la Web como recurso lingüístico para llevar a cabo tareas relacionadas con el procesamiento del lenguaje natural.

3.4 Uso de la Web como corpus

La Web es un medio para acceder de manera rápida y fácil a una gran variedad de información almacenada en formato electrónico en diferentes partes del mundo. Hoy por hoy la Web es el repositorio más grande de información que existe. La Web es inmensa y contiene cientos de billones de palabras de texto; además, es gratis y disponible con un clic del ratón. El uso de la Web como corpus ha tenido gran interés en la última década, principalmente porque se puede aplicar a diversas tareas del procesamiento del lenguaje natural (Ferraresi et al., 2008; Kilgarriff, 2003).

Se estima que el tamaño actual de la Web en septiembre del 2009 era de 21.61 billones de páginas indexadas por los diferentes motores de búsqueda[1]. Como se puede apreciar, la cantidad de documentos en formato electrónico que se encuentran en la Web es muy grande y en varios idiomas y precisamente de ahí el interés de usar la Web como recurso lingüístico; de hecho este interés ha llevado a desarrollar sesiones especiales dentro de las conferencias del área, donde el tema central es el uso de la Web como corpus[2]. Además, la Web es multilingüe, ya que aproximadamente: 71% de las páginas están escritas en el idioma inglés, 6.8% en japonés, 5.1% en alemán, 1.8% en francés, 1.5% en chino, 1.1% en español, 0.9% en italiano, y el 0.7% en sueco; el restante 11.1% está repartido en otros idiomas y dialectos con porcentajes de presencia menores (Kilgarriff, 2003).

1 Las cifras más actualizadas se encuentran disponibles en http://www. worldwidewebsize.com, donde además, se puede obtener la información del tamaño de la Web por país

2 http://www.sigwac.org.uk/wiki/WAC5

Una clasificación, respecto al tipo y tamaño de archivos (ppt, pdf, doc, etc.) de la Web es presentada por Mihalcea (Mihalcea et al., 2006) quien utiliza esta información para facilitar la construcción de corpus a partir de la Web para su uso en tareas del procesamiento del lenguaje natural. Adicionalmente se han desarrollado sistemas que permiten obtener un corpus a partir de la Web, por ejemplo, Beroni (Beroni et al., 2004) introduce un método en la que dada una lista de palabras clave de un dominio, a las que llaman semillas, se recolecta información relevante de la Web haciendo combinaciones entre estas semillas. Chakrabarti (Chakrabarti et al., 1999) presenta un sistema que permite llevar a cabo el filtrado de información proveniente de la Web utilizando "topic classifier" y con esto pretender recuperar información que tenga una alta relación con un dominio determinado.

Existen varios trabajos de investigación en los cuales se ha utilizado la Web como corpus, aunque la mayoría sólo utilizan la Web para buscar ejemplos de uso de alguna petición en particular. Esta información es utilizada ya sea como corpus de entrenamiento (Cavagliá et al., 2001), o bien como complemento a un conjunto de ejemplos etiquetados (Volk, 2002). Kilgarriff presenta un sistema de consulta[3] que permite la formación de corpus a partir de la Web donde, además, se pueden obtener relaciones gramaticales, así como distribuciones tipo tesauro. El corpus es formado a partir de un resumen gramatical de las palabras utilizadas, así como del comportamiento de las colocaciones dadas.

Sin embargo, la Web tiene varios aspectos negativos, entre ellos, el que es muy heterogénea y desorganizada, además existe mucha información basura o con etiquetas que dificultan su procesamiento. Aunado a que no se puede estar seguro de que todo lo encontrado esté correcto, ya que nadie lo revisa. Pero gracias a la redundancia de la Web la información correcta suele prevalecer.

3 http://www. sketchengine.co.uk

En investigaciones recientes llevadas a cabo en el centro de investigación de la Web[4], se estudia el gran potencial el uso de la Web en actividades del procesamiento del lenguaje natural, entre estas actividades se encuentra el uso de la Web como corpus en la desambiguación del sentido de las palabras. Además existen trabajos, como el desarrollado por Resnik (Resnik et al., 2000), en el cual, para una oración dada, es generado un análisis sintáctico.

Los resultados del análisis sintáctico son utilizados como patrón de búsqueda en la Web con la finalidad de obtener ejemplos con la estructura especificada. El contexto puede ser revisado tanto en snippets (pequeña porción de texto proporcionada por una máquina de búsqueda como respuesta a una petición) como en documentos, a partir de los contextos se realiza un análisis de correlación. De esta manera los autores muestran que pueden ser probadas reglas sintácticas con los datos obtenidos de la Web.

En (Rosso et al., 2005), se presenta una aproximación que utiliza la Web para la desambiguación del sentido de las palabras, la cual utiliza la información de la Web de manera directa, en vez de extraer ejemplos de entrenamiento. Específicamente se hace un conteo de los pares hiperónimo-adjetivo e hipónimo-adjetivo considerando todos los posibles sentidos de un sustantivo.

Blum y Mitchell utilizaron el aprendizaje semi-supervisado, específicamente el co-training, para categorizar páginas Web utilizando un sistema de categorización basado en naïve Bayes partiendo de un texto de entrenamiento proporcionado y la página Web como dos fuentes de información sobre un mismo caso (Blum et al., 1998). En relación con la categorización de páginas Web, se han desarrollado otros trabajos tales como (Govert, 1999) en el cual, el autor se basa en una descripción probabilística que permite representar los documentos bajo estudio, para llevar a cabo la categorización de los documentos utiliza el método de los k-vecinos mas cercanos. Por otro lado en (Apte et al., 1994) generan automáticamente reglas de decisión que permiten completar las reglas hechas por anotadores humanos, con la finalidad de incrementar la exactitud de categorización.

4 http://www.ciw.cl/

Existen además trabajos como el de Veronis (Veronis, 2004) donde presenta una herramienta para determinar el uso de una palabra en base texto mostrando coocurrencias de la palabra por medio de vectores. Una descripción actualizada de los distintos métodos y aproximaciones enfocados a la tarea de la desambiguación del sentido de las palabras se encuentra en (Navagli, 2009).

Estos métodos que utilizan la Web para la tarea de categorización tienen la desventaja de formar una bolsa de ejemplos no etiquetados de la cual son seleccionados los que se incorporarán al conjunto de entrenamiento, pero no llevan una categoría previa a la hora de hacer la descarga. Tampoco siguen un proceso de formación de peticiones minucioso, es decir, que permitan la incorporación de nuevas instancias al sistema de categorización de manera que permita incrementar las diferencias entre las categorías.

En esta tesis se propone un método general de categorización textual, el cual utiliza la Web como recurso lingüístico para llevar a cabo la tarea de categorización automática de documentos de manera mínimamente supervisada. Dentro de las razones más importantes para usar la Web como corpus lingüístico está el poder acceder de manera rápida y fácil a una gran variedad de información almacenada en formato electrónico en diferentes partes del mundo.

3.5 Categorización de documentos con clases desbalanceadas

La aplicación de técnicas de aprendizaje computacional a problemas reales ha traído consigo una serie de retos que previamente no se habían considerado relevantes. Uno de estos problemas es el de las categorías desbalanceadas; éste se presenta cuando una categoría tienen significativamente más instancias que otra. Este tema ha cobrado recientemente gran interés en la comunidad científica (Chawla et al., 2004) debido a que el problema de las clases desbalanceadas es relativamente común a un gran cantidad de aplicaciones reales, y que los algoritmos actuales de aprendizaje tienen desempeños pobres para la clase minoritaria, la cual, en general, es justamente la que más nos importa categorizar correctamente (Maloof, 2003). Algunos ejemplos de aplicación en los cuales se presenta este problema se enlistan a continuación:

- Detección de fraudes (la mayoría de las personas no los comente) (Vhan et al., 1998)
- Diagnóstico de fallos (ocurren en pocas ocasiones) (Weiss et al., 1998)
- Diagnóstico médico (la mayoría de la personas son sanas) (Grzymala-Buse et al., 2000)
- Percepción remota (existen relativamente pocas imágenes con la percepción de interés) (Kubat et al., 1998).

Los algoritmos de categorización buscan en general regularidades en los datos que expliquen la mayor cantidad de ejemplos. También en general tienden a minimizar el error global (Mitchell, 1997). El contar con pocos datos para cada clase, dificulta el desempeño de los sistemas de categorización por que existen pocos datos para soportar los posibles patrones que se van construyendo. Si se tienen pocos datos en la categoría minoritaria y están relativamente dispersos, es muy difícil para un sistema de categorización encontrar patrones que los cubran con el suficiente soporte estadístico como para considerarlos. Este poco soporte provoca que con las métricas utilizadas no se construyan patrones que cubran muy pocos ejemplos.

El modificar métricas con poco soporte representa el inconveniente de confundir el posible ruido en los datos con los datos validos. Para ilustrar este hecho más claramente, supongamos que tenemos 1000 datos, 30 de los cuales pertenecen a la categoría minoritaria y el resto son de la clase mayoritaria. Es relativamente común para un sistema de categorización inducir una categorización trivial que predice todo como la categoría mayoritaria, con la cual se obtendría una exactitud del 97%, aunque se equivoque en todos los ejemplos de la clase minoritaria.

Han sido propuestas un gran número de soluciones para el problema de las categorías desbalanceadas tanto a nivel de datos como a nivel algoritmo. A nivel de datos, esas soluciones incluyen diferentes formas de muestreo. Estos muestreos pueden ser en forma aleatoria o bien en forma avanzada, las cuales se dividen en: sub-muestreo y sobre-muestreo aleatorios también conocidos como *over-sampling* y *under-sampling,* respectivamente (Japkowicz et al., 2002). A nivel de algoritmo, las soluciones incluyen ajustes de costos de varias categorías (Eibe et al., 2006) y ajuste de estimación probabilística para los atributos finales (Chawla et al., 2004). A continuación se definen los métodos utilizados para resolver el desbalanceo en las categorías a nivel de datos.

3.5.1 Método de sub-muestreo

El método de sub-muestreo (under-sampling), consiste en eliminar aleatoriamente elementos de la o las categorías mayoritarias hasta obtener el mismo tamaño que la categoría minoritaria (Batista et al., 2004). Una estrategia de sub-muestreo avanzada consiste en eliminar solo ejemplos de la categoría mayoritaria que sean redundantes o muy cercanos con los de la categoría minoritaria (Kubat et al., 1997).

La implementación tradicional de los métodos para realizar el sub-muestreo contempla la construcción de muestras del mismo tamaño que el conjunto original (Dietterich, 1997). El tamaño de las muestras resultantes es importante en el sentido que, cuando se utilizan muestras de poco tamaño se requieren pocos recursos computacionales (en tiempo y espacio de almacenamiento) para su tratamiento, sin embargo se corre el riesgo de desechar información importante que permita distinguir una categoría otra.

3.5.2 Método de sobre-muestreo

En este método (over-sampling), existen dos variantes: la primera conocida como sobre-muestreo aleatorio la cual consiste en generar ejemplos de la categoría minoritaria aleatoriamente hasta tener tantos ejemplos como la otra categoría. La segunda variante se conoce como sobre-muestreo enfocado, y consiste en generar ejemplos de la categoría minoritaria aleatoriamente, pero limitando a cierto número las nuevas instancias.

Otra de las técnicas avanzadas de sobre-muestreo es conocida como SMOTE, la cual crea nuevos ejemplos de la categoría minoritaria interpolando los valores de vecinos más cercanos a ejemplos de la categoría minoritaria (Chawla et al., 2004). También existen algunas estrategias que combinan el sub-muestreo con sobre-muestreo (Batista et al., 2004).
Tanto el sub-muestreo como el sobre-muestreo, llevados a cabo de manera aleatoria, presentan los siguientes inconvenientes:

(i) El sub-muestreo aleatorio puede eliminar ejemplos de la categoría mayoritaria potencialmente útiles para el proceso de aprendizaje (Japkowicz et al., 2002).

(ii) El sobre-muestreo aleatorio repite ejemplos de la categoría minoritaria buscando que sean incluidos en los patrones del sistema de categorización, sin embargo, pueden repetir ejemplos con ruido y en general provoca un sobre-ajuste (overfitting) en los modelos de aprendizaje generados (Domingos, 1999).

3.6 Categorización no temática

Existen una serie de tareas que se pueden llevar a cabo dentro del área de categorización no temática de documentos, entre ellas destacan:

- Atribución de autoría: decidir si un texto fue escrito por un cierto autor o no; (Koppel et al., 2004)
- Detección de Plagio: encontrar similitudes entre dos textos; (Meyer et al, 2007; Barron et al., 2009)
- Perfil de autor o caracterización: extraer información acerca de la edad, educación, sexo, etc., del autor de un texto; (Koppel et al., 2002)
- Detección de inconsistencias de estilo: determinar cómo puede ocurrir en la redacción en colaboración; (Graham et al, 2005)
- Clasificación de opiniones y análisis de sentimientos: clasificar opiniones a favor o en contra sobre algún tópico en particular, por ejemplo películas; (Grieve, 2007; Banea et al., 2008)

Dentro de toda esta gama de tareas, no exhaustiva, se presenta como problema de categorización no temática el típico problema de la atribución de autoría, en el cual un texto de autor desconocido es asignado a un autor, dentro de un conjunto de autores disponibles. Desde el punto de vista del aprendizaje automático, esto puede ser visto como un problema de categorización multi-categorías (Sebastiani, 2002). Esta tarea se describe a detalle en al siguiente sección. En el capítulo 5 se presentan los resultados obtenidos aplicando el método propuesto en esta tesis para abordar este problema.

3.6.1 Atribución de autoría

El estilo es la manera en que una persona actúa, lo cual marca la acción en sí misma con una inscripción única. Para un experto, su estilo es una forma natural de ejecutar una actividad cualquiera. Por ejemplo, un escritor después de un periodo de duro trabajo y experimentación, desarrolla su propio estilo. En general, cada persona adquiere su propio estilo a lo largo de su vida, de acuerdo a las experiencias que ha tenido en su vida laboral, intelectual y personal.

En nuestro caso, limitándonos al estilo de un texto, adoptaremos la siguiente definición: *"estilo es el conjunto de distintos aspectos o rasgos que caracterizan la escritura de un texto"* (Alcaraz et al., 1997). Estos rasgos son los que identifican a un documento como elemento de una categoría. Desafortunadamente, estos rasgos no son claros y en algunos casos desconocidos; por ejemplo, si nos preguntamos:

¿*Cuáles son los rasgos que identifican a los textos escritos por Octavio Paz*?

No podríamos responder de manera clara y concisa, ya que el estilo no es sólo un ingrediente en una pieza de escritura, sino un reflejo que proviene desde el autor. Es decir, es dependiente de cómo el autor transfiere una idea a lenguaje escrito. En consecuencia, el escrito final está fuertemente relacionado al autor. En otras palabras, el estilo es indicado por características que revelan la elección del autor de un modo de expresión, es decir, la elección específica de palabras, estructuras sintácticas, estrategias del discurso o combinaciones.

Además, existen variaciones que influyen en el texto tales como el grado o nivel de estudios, el estatus social, la personalidad del autor, la audiencia y la época en que fue escrito el documento. Todas estas variaciones son independientes de la temática del texto, pero son determinantes para la correcta categorización de dicho texto.

Existen varios factores que influyen para categorizar correctamente un texto y más aún, hay una infinidad de rasgos que caracterizan el estilo de escritura de un autor y que son difíciles de identificar, debido a que el estilo de un autor se refiere a las particulares condiciones de apropiación y actualización de los enunciados. Sin embargo, éste puede ser variable debido a la diferencia en temas o género y también al desarrollo de cada autor a través del tiempo.

La principal tarea en la atribución de autoría es identificar las características, las cuales deben ser invariantes, que pueden ayudar a discriminar a un autor de otros. En contraste con las tareas de categorización temática, en este caso no es claro cómo determinar el conjunto de características que deben ser utilizadas para identificar un autor. Así, el reto principal de esta tarea es la caracterización apropiada de los documentos que capturen el estilo de escritura de los autores.

En resumen, la atribución de autoría es un problema donde un conjunto de documentos con autores conocidos es utilizado como entrenamiento de modelos para posteriormente determinar automáticamente el autor de un texto anónimo, esto es, no visto por el conjunto de entrenamiento. De esta manera, sólo capturamos rasgos relevantes de un autor que permiten identificarlo contra un conjunto de autores determinado.

Existen trabajos que tratan de identificar el estilo de un autor en diferentes contextos. Por ejemplo, en la categorización de correos electrónicos (de Vel O et al., 2001; Argamon et al., 2003), en la detección de plagio (Zhao, 2005; Barron et al., 2009) o en el análisis forense de un texto, que intenta determinar el autor en relación a una investigación criminal (de Vel O et al., 2001).

A continuación se presentan los trabajos centrales que se han realizado con la temática de categorización de textos por estilo. Este panorama general se orienta a los métodos de caracterización que se han propuesto en la literatura, principalmente los utilizados en atribución de autoría (Stamatatos, 2009a).

- ***Caracterización estilométrica***: Los primeros intentos en la caracterización de documentos por estilo provienen de esfuerzos de análisis literarios. Básicamente éstos se enfocaron exclusivamente en el uso de medidas estilométricas. Características como la longitud de las palabras o de las oraciones, así como la riqueza del vocabulario han sido algunas de las medidas utilizadas (Corney et al., 2002). A pesar de que intuitivamente la amplitud del vocabulario y la frecuencia de uso de tales palabras parecerían ser elementos básicos característicos de cada autor, este tipo de características no son suficientes. Al parecer esto se debe a que, por un lado, existen importantes variaciones aún para el mismo autor dependiendo del tipo de texto; y, por otro lado, estas características son en extremo sensibles al tamaño del documento, perdiendo gran parte de su significado para textos pequeños. Ejemplos de estas medidas pueden observarse en la figura 3.1.

Promedio = número de palabras / número de oraciones ***Riqueza del vocabulario*** = número de palabras / total del vocabulario ***Hápax*** = número de palabras cuya ocurrencia en el documento es uno / total de vocabulario ***Riqueza de palabras*** = número de oraciones / número de palabras ***Riqueza de oraciones*** = número de oraciones / total del vocabulario ***Palabras en mayúsculas*** = (palabras que comienzan con mayúsculas – número de oraciones) / número de oraciones ***Promedio de las palabras*** = total de caracteres / número de palabras

Figura 3.1: Ejemplos de características estilométricas

- ***Caracterización sintáctica:*** Otro intento es la caracterización de los textos por un conjunto de marcadores de estilo (*style markers*). Por ejemplo, el uso del impersonal para caracterizar documentos técnicos. Estos marcadores de estilo van más allá de las simples medidas estilométricas sobre las palabras e integran información sobre la estructura del lenguaje empleado.

  Para ello, es necesario realizar un análisis complejo usando analizadores morfológicos y sintácticos (i.e., taggers, parsers) (Chaski, 2005; Finn et al., 2003). Así, un texto se caracteriza por la presencia y frecuencia de ciertas estructuras sintácticas. Desafortunadamente, esta caracterización es costosa y en algunos casos imposible dada la inexistencia de tales herramientas para el idioma en cuestión.

- ***Caracterización léxica***: Este enfoque incluye por lo menos tres métodos diferentes. En el primero, la caracterización se realiza usando exclusivamente un conjunto de palabras de paro (stopwords), ignorando las palabras de contenido (Argamon et al., 2003; Argamon et al., 2005; Stamatatos et al., 2000; Zhao et al., 2005). A pesar de que las palabras de paro no parecen ser marcas de estilo confiables, ya que son muy frecuentes y ocurren en todo texto, el uso y frecuencia de estas palabras es característico del estilo de los autores. La categorización basada en este tipo de caracterización trabaja apropiadamente pero es muy sensible al tamaño de los documentos. En este caso, la longitud de los documentos no sólo influye en la frecuencia de ocurrencia de las palabras de paro sino, también, en su posible presencia.

  Un segundo método usa la representación tradicional de bolsa de palabras considerando únicamente las palabras de contenido (Diederich et al., 2003; Finn et al., 2003; Keselj et al., 2003), es decir, el enfoque tradicional usado para la categorización temática. Este método produce resultados aceptables siempre y cuando exista una fuerte correlación entre los temas y los autores.

  Finalmente, un tercer método considera *n*-gramas, es decir, secuencias de *n* palabras sucesivas. Este método intenta capturar la estructura del lenguaje de los textos por medio de simples secuencias de palabras en contraste de las complejas estructuras sintácticas (Fürnkranz, 1998; Keselj et al., 2003; Peng et al., 2004; Stamatatos, 2009b).

  De esta manera, el propósito es obtener una adecuada caracterización de los textos sin llevar a cabo un costoso análisis sintáctico. Desafortunadamente este tipo de caracterización nos lleva a una explosión combinatoria, por lo que, comúnmente, se emplean secuencias de una, dos o a lo más tres palabras (uni-gramas, bi-gramas y tri-gramas). En conclusión, los n-gramas permiten un mejor tratamiento de las frases como “Bill Gates” o “White House”, debido a que la representación anterior (*bolsa de palabras*) separaba estas palabras y su estructura se pierde.

El enfoque propuesto en este trabajo se ubica en este último esquema, limitando la representación de los documentos a su nivel léxico. Esto se debe a que se trata de una caracterización simple, pero general, la cual puede aplicarse a cualquier tipo de texto, sin importar su dominio e incluso el idioma. El principal problema de esta caracterización es que cada documento será representado por un enorme número de atributos. En el capítulo cinco se muestran los resultados experimentales obtenidos en esta tarea. Sin embargo, es pertinente comentar cómo se podría comparar el enfoque propuesto en el presente trabajo de tesis con las investigaciones anteriores que trabajan a nivel léxico usando n-gramas en la tarea de atribución de autoría. Esta comparación se podría llevar a cabo uniendo atributos que permitan identificar a un determinado autor. Por ejemplo, uniendo uni-gramas más bi-gramas o uni-gramas más bi-gramas más tri-gramas y utilizar el conjunto resultante como conjunto de entrenamiento. Así como se detalla en el capítulo cinco, esto es lo que se hace en el presente trabajo. Aunque no se lleva a cabo la comparación con otros métodos, al no existir colecciones estándar de evaluación, bien podría ser considerado como una actividad futura del presente trabajo de investigación.

3.7 Conclusiones del capítulo

En este capítulo se presentaron diversos trabajos relacionados con distintas tareas de categorización automática de textos. La idea de este capítulo fue presentar no solo los problemas de investigación abiertos actualmente sino, además, las soluciones propuestas en el estado de arte a estas problemáticas. En la presente tesis se presenta el desarrollo de un sistema que permite soportar tareas de categorización tanto temática como no temática utilizando la Web como recurso lingüístico. El método de clasificación automática basado en self-training y en el uso de la Web como corpus que se propone en el siguiente capítulo incluye un arreglo stacking, el cual es responsable de la selección de los snippets descargados de la Web que serán incorporados al conjunto de entrenamiento. En nuestro caso el ensamble está formado por un sistema de categorización basado en naïve Bayes y otro en SVM. Los snippets seleccionados son aquellos que de manera conjunta les es asignada la probabilidad más alta.

Sin duda alguna la construcción de un corpus de entrenamiento, para cualquier tarea, es muy laboriosa. Existen varios trabajos de investigación en los cuales se utiliza la Web como corpus, aunque generalmente se usa solamente para extraer más ejemplos de entrenamiento. En el caso presentado en el siguiente capítulo además de esto se pretende hacer la extracción de patrones, que permitan llevar a cabo la tarea de categorización, directamente de la Web. Se ha utilizado la Web como corpus para llevar a cabo tareas de categorización automática de documentos y de desambiguación del sentido de las palabras, cuyos resultados se presentan en los capítulos 4 y 5 respectivamente.

Capítulo IV

Método propuesto

En este capítulo se presenta el método semi-supervisado para la categorización automática de documentos propuesto en este trabajo de tesis. Este método funciona con pocos ejemplos etiquetados como entrenamiento e incorpora gradualmente información no etiquetada descargada de la Web. La finalidad de incorporar ejemplos no etiquetados al conjunto de entrenamiento es la de mejorar la exactitud a la hora de llevar a cabo la tarea de categorización automática de documentos. Se presenta, además, una descripción de cada una de las partes que lo componen. También se describe el proceso para incorporar ejemplos no etiquetados al conjunto de entrenamiento desde la Web, aunque esto no es limitativo a la Web y en realidad se puede utilizar cualquier colección de datos no etiquetados, siempre y cuando sea suficientemente grande como para asegurar la presencia de ejemplos pertinentes para su integración al conjunto de entrenamiento.

4.1 Introducción

Así como se mencionó en los capítulos anteriores, las aproximaciones supervisadas tradicionales para llevar a cabo la categorización automática de documentos, consideran un gran número de instancias etiquetadas en la fase de entrenamiento. De hecho, entre más y mejores ejemplos de entrenamiento tenga un sistema de categorización automática de documentos basado en aprendizaje supervisado, mejores resultados tendrá. Este hecho es una limitación en el uso de estas aproximaciones ya que frecuentemente, las instancias etiquetadas, son difíciles de obtener debido a que requieren ser etiquetadas manualmente, lo cual hace que el proceso sea muy tardado y costoso. Por otro lado, los datos no etiquetados son muy fáciles de obtener pero sólo pueden ser utilizados en pocas aplicaciones de manera efectiva.

En este escenario el aprendizaje semi-supervisado emerge como la solución a este problema, debido a que considera el uso de pocas instancias manualmente etiquetadas como entrenamiento y la incorporación al mismo de datos no etiquetados con la finalidad de mejorar la exactitud de un sistema de categorización automática de documentos. En el presente trabajo de tesis se desarrolló un método semi-supervisado de categorización automática de documentos, el cual requiere de muy pocos ejemplos etiquetados y de manera gradual e iterativa incorpora ejemplos no etiquetados con la finalidad de mejorar la precisión de categorización. Los ejemplos no etiquetados son descargados de la Web. En las siguientes secciones se describe la arquitectura y las etapas que componen este método.

4.2 Arquitectura

En este trabajo se presenta un método de categorización automática de documentos basado en aprendizaje semi-supervisado. En particular se propone un método de categorización basado en la técnica de self-training. Este método utiliza un pequeño conjunto de documentos etiquetados, los cuales son utilizados para entrenar al sistema. El conjunto de entrenamiento es enriquecido incorporando ejemplos no etiquetados. Los ejemplos no etiquetados son descargados de la Web, pasando previamente por un proceso de selección que permite la incorporación sólo de las mejores instancias no etiquetadas al conjunto de entrenamiento. La finalidad de incorporar información no etiquetada al conjunto de entrenamiento es mejorar la precisión del sistema de categorización.

La figura 4.1 muestra la arquitectura general del método propuesto, el cual consiste en dos procesos principales:

(i) *adquisición de corpus de la Web* y

(ii) *aprendizaje semi-supervisado.*

Estos dos procesos son complementarios e independientes ya que aunque en este trabajo utilizamos ejemplos no etiquetados descargados de la Web, esto no es condición ya que con cualquier conjunto de datos no etiquetados el método funcionaria exactamente igual.

La principal diferencia del método descrito en el presente capítulo para llevar a cabo la tarea de categorización automática de documentos con respecto a otros métodos existentes en el estado del arte es que no requiere de un conjunto previo de instancias no etiquetadas. El método propuesto va a la Web y obtiene los ejemplos no etiquetados. Para esto se lleva a cabo todo un proceso de formación de peticiones cuya finalidad es la de asociar lo más posible la petición con la clase a la que pertenecen los ejemplos a partir de la cual se forma la petición. Esto podría ser considerado como una categoría previa de la información descargada de la Web. En los párrafos siguientes se describe más a detalle este proceso.

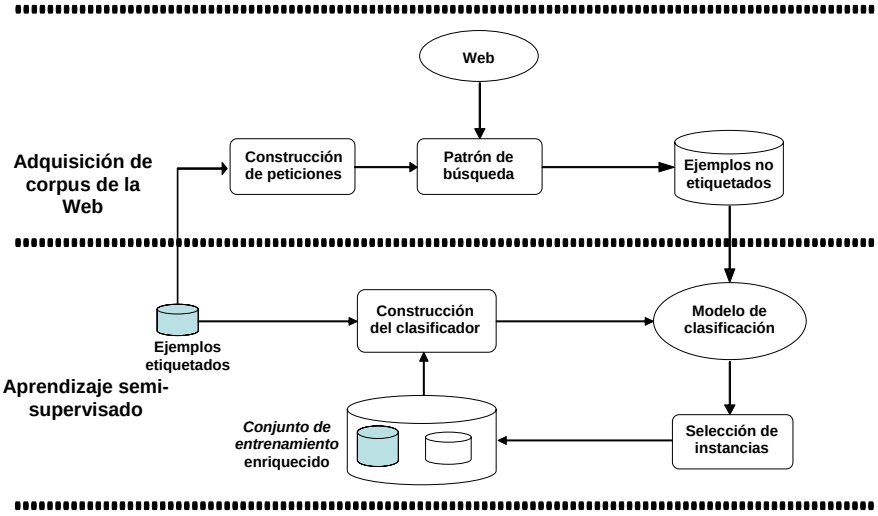

Figura 4.1: Método semi-supervisado de categorización basado en la Web

A continuación se presentan el objetivo y una descripción general de los procesos que componen el método propuesto:

- *Adquisición del corpus de la Web:* El objetivo de esta etapa es descargar ejemplos no etiquetados de la Web. Estos ejemplos son descargados para cada categoría de manera independiente; para llevar a cabo esta tarea se construyen una serie de peticiones formadas por las palabras relevantes de cada categoría, es decir, aquellas palabras que permiten distinguir a una categoría de otra. De esta manera, los ejemplos no etiquetados descargados de la Web llevan asociada una categoría previa asignada por la petición utilizada para realizar la descarga[1].

- *Aprendizaje semi-supervisado* (self-training): La finalidad de esta etapa es incrementar el tamaño del conjunto de entrenamiento para mejorar la precisión en la categorización.

1 En este trabajo utilizamos el corpus formado a partir de la Web. Sin embargo, el método funciona igual si se proporciona una colección de ejemplos no etiquetados.

Específicamente, los ejemplos no etiquetados descargados en el punto anterior pasan por un proceso de selección, el cual es el responsable de seleccionar sólo las mejores instancias no etiquetadas para ser incorporadas al conjunto de entrenamiento, es decir, aquellos ejemplos no etiquetados que permitan incrementar las diferencias entre las categorías. El método propuesto utiliza dos fuentes de información complementarias para predecir la categoría de un ejemplo no etiquetado: es decir, por un lado la categoría "previa" asignada por la Web y por otro lado la categoría asignada por la etapa de aprendizaje semi-supervisado. Este proceso depende del número de instancias etiquetadas que se tengan, ya que entre más instancias de entrenamiento se tengan, se pueden formar mejores peticiones y de esta manera tener ejemplos no etiquetados que tengan una mayor relación con la categoría a la que pertenece la petición.

Este método resulta muy adecuado para realizar trabajos de categorización de documentos cuando se tienen muy pocos ejemplos de entrenamiento ya que, como se describe a lo largo de este capítulo, se pueden llevar a cabo experimentos utilizando desde un sólo ejemplo de entrenamiento etiquetado por categoría. Ésta es una gran ventaja del método propuesto, ya que a partir de un número muy reducido de instancias etiquetadas puede hacer crecer el conjunto de entrenamiento, logrando incrementar la exactitud de categorización. En el siguiente apartado se presenta una descripción de las partes que componen a la arquitectura propuesta, la cual como se pudo apreciar en la figura 4.1, se divide en dos partes principales: adquisición de corpus de la Web y aprendizaje semi-supervisado.

4.2.1 Adquisición de corpus

Este proceso considera la extracción automática de ejemplos no etiquetados de la Web. Para llevar a cabo esta tarea, primero se construyen una serie de peticiones combinando las palabras más significativas de cada una de las categorías. Posteriormente, estas peticiones son lanzadas a la Web para descargar ejemplos no etiquetados, relacionados con la clase correspondiente. A continuación se describe el proceso para formar las peticiones.

- ***Construcción de peticiones***: Para construir las peticiones, es necesario, primeramente, determinar el conjunto de palabras relevantes para cada una de las clases que forman el conjunto de entrenamiento. El criterio utilizado para este propósito está basado en dos medidas de las palabras dadas: por un lado, la *frecuencia de ocurrencia*, y por otro lado, la *ganancia de información* (Guzmán et al., 2007). Se consideraron estas dos medidas debido a que la medida de frecuencia está asociada con la clase a la cual pertenece la palabra, mientras que la medida de ganancia de información involucra a todas las categorías. De esta manera tenemos, por un lado, las palabras más relevantes a una categoría, y por otro, las palabras que nos permitirían separar mejor a una categoría de otra. A este conjunto de palabras les llamamos *palabras relevantes*. Específicamente, consideramos que una palabra w_i es relevante a la categoría C si:

    1. La frecuencia de ocurrencia de w_i en C es mayor que el promedio de ocurrencia de todas las palabras. Cabe mencionar que sólo son consideradas aquellas palabras que tienen una frecuencia mayor que 1 en la categoría, esto es:

$$f_{w_i}^{C} > \frac{1}{|C'|} \sum_{\forall w \in C'} f_w^C \text{, donde } C' = \{ w \in C \mid f_w^C > 1 \}$$

2. La ganancia de información de w_i en el conjunto de entrenamiento es positiva ($IG_{w_i} > 0$). Esta medida complementa a la anterior, ya que la ganacia de información involucra a todas las categorías. Esto intuitivamente nos permite seleccionar las palabras que representan a una categoría y que por lo tanto nos permiten diferenciar una categoría de otras. Formalmente se obtiene un conjunto de palabras asociadas a la categoría en cuestión, cada una de estas palabras lleva asociado un peso. En este trabajo consideramos sólo las palabras que tienen una ganancia de información positiva.

Con la aplicación de estas medidas se pretende seleccionar las palabras con más alto valor descriptivo, medida de frecuencia, pero también las palabras con el más alto nivel discriminativo, medida de ganancia de información. Esto es, aquellas palabras que nos permitan distinguir una categoría de otra y con las cuales podamos recuperar fragamentos de textos similiares en los ejemplos no etiquetados descargados de la Web.

Aplicando estas medidas de manera conjunta, una palabra que ocurra en diferentes categorías tendrá diferente peso y, por lo tanto, diferente impacto en la cantidad de información que con ella se descarga. Cabe mencionar que existe un pesado tradicional que hace algo parecido y que se describió en el capítulo 2: el pesado $tf \cdot idf$ (Salton et al., 1987). Este pesado, a diferencia del nuestro, está pensado para contrarrestar el efecto negativo que tiene el hecho de que una palabra sea muy frecuente, pero se encuentre en muchos documentos. Para visualizar mejor la diferencia, supongamos que una palabra aparece en más de una categoría como palabra frecuente. Esta palabra tendrá diferente peso en función del valor que se asocie por medio de la ganancia de información a dicha palabra y por lo tanto diferente impacto en la cantidad y calidad de información no etiquetada que con ella se descargará de la Web. Mientras que en esta misma situación utilizando el pesado $tf \cdot idf$ la palabra perdería peso al encontrarse en muchos documentos y distintas categorías, lo cual en nuestro caso implicará recuperar informacion no etiquetada de menor calidad.

Una vez obtenido el conjunto de palabras relevantes para cada una de las categorías, es posible llevar a cabo la construcción del correspondiente conjunto de peticiones. El número de peticiones por categoría será variable y dependerá del número de palabras que superen las medidas anteriores. El objetivo de utilizar los filtros de frecuencia y ganancia de información es que contemos con las palabras más representativas de cada categoría, con lo cual esperamos aumentar la relevancia de las páginas recuperadas.

A manera de ejemplificar el proceso llevado a cabo por el método, se presenta en la tabla 4.1 el conjunto de las primeras diez palabras relevantes para la categoría *wheat* de la colección Reuters[2]. La descripción completa de esta colección, así como los resultados experimentales obtenidos se describen en el capítulo 5. Para cada una de estas palabras se muestra la frecuencia y la ganancia de información así como el peso de la palabra (*f*IG*). Las palabras se encuentran ordenadas de acuerdo a la ganancia de información.

Tabla 4.1: Palabras relevantes para la categoría *wheat*

Palabra	*f*	*IG*	*f*IG*
wheat	418	199.34	83325.87
grain	164	116.65	19131.12
tonnes	210	55.56	11668.86
corn	89	39.51	3516.92
agriculture	75	18.52	1389.37
trade	42	12.17	511.20
export	76	10.67	810.95
usda	54	7.64	412.61
crop	51	5.97	304.57
washington	44	5.79	255.16

2 http://www.daviddlewis.com/resources/testcoleections/Reuters21578

Se puede observar que para determinar el peso de las palabras tanto la frecuencia como la ganancia de información juegan un papel muy importante ya que las dos están directamente involucradas. Por ejemplo, en las palabras *trade* y *export* coincide que *trade* tiene mayor ganancia de información que *export*, pero menor frecuencia lo que hace que *export* tenga un peso mayor. Como se mencionó anteriormente el peso de una palabra tiene una repercusión directa en la cantidad de información que con ella se descargará y en este caso será mayor la cantidad de información que se descargará con *export* a pesar de tener un valor de ganancia de información menor.

Las peticiones que se lanzan a la Web consisten en las combinaciones de tamaño tres ($_nC_3$) que se obtienen de las *n* palabras que superaron los filtros de frecuencia y ganancia de información de cada categoría. Se eligieron peticiones de este tamaño debido a que en algunos trabajos encontrados en el estado del arte, tales como en (Zelikovits et al., 2006), llevan a cabo experimentos utilizando peticiones de dos palabras como patrón de búsqueda y reportan obtener baja precisión en la relevancia de las páginas recuperadas. Básicamente argumentan que la ambigüedad de las palabras usadas en la petición afecta la calidad de las páginas recuperadas. Considerando este argumento y tomando en cuenta que al hacer la petición en la máquina de búsqueda se lleva a cabo un *and* lógico, si fijamos en tres el tamaño de la petición, aumentamos la probabilidad de que la página recuperada sea relevante, mientras que si utilizamos más de tres palabras en la petición puede ocurrir que el número de páginas recuperadas disminuya significativamente por hacer la petición muy específica a un dominio determinado.

Con la finalidad de ejemplificar la formación de peticiones utilizando el método propuesto, en la tabla 4.2 se presentan diez peticiones generadas con el conjunto de palabras relevantes mostradas en la tabla 4.1. En total, para el conjunto de diez palabras relevantes, se forman 120 peticiones ($_{10}C_3 = 120$).

Tabla 4.2: Algunas peticiones para la categoría *wheat*

Petición
wheat grain tonnes
wheat grain agriculture
wheat grain export
wheat grain crop
wheat tonnes corn
wheat tonnes trade
wheat tonnes usda
wheat tonnes washington
grain tonnes export
grain agriculture trade

Como se puede observar en la tabla 4.2 una misma palabra puede ocurrir en varias peticiones. Sin embargo entre una petición y otra siempre habrá al menos una palabra distinta, con lo cual las probabilidades de recuperar información distinta de la Web se incrementan, ya que la máquina de búsqueda utilizada, en este caso Google, realiza una operación lógica *and* con las palabras utilizadas en la petición, de esta manera, al tener palabras distintas entre las peticiones, se accedera a páginas distintas. También podemos observar en la tabla 4.2 la relación que tienen las palabras que forman la petición con la clase a la que pertenecen, en este caso la clase *wheat*, lo que intuitivamente nos lleva a pensar que la información que se recuperará con estas peticiones tiene altas posibilidades de pertenecer a esta categoría.

Cada palabra tiene un peso asociado, lo que ocasiona que la petición en sí tenga un peso. El peso de una petición $q = \{w_1\ w_2\ w_3\}$ para una determinada categoría C, está determinado por:

$$\Gamma_C(q) = \sum_{i=1}^{3} f_{w_i}^{C} \times IG_{w_i}$$

Por la manera como son formadas las peticiones con el conjunto de palabras relevantes, utilizando los criterios de frecuencia y ganancia de información, un incremento en el número de ejemplos de entrenamiento no significa necesariamente un incremento en el número de peticiones por categoría.

Para seguir con la ejemplificación del método, en la tabla 4.3 se muestra el peso asociado a cada una de las peticiones mostradas en la tabla 4.2. Como se puede apreciar, la suma de los pesos de cada palabra es lo que determina el peso de la petición y, como se verá a continuación, la cantidad de información descargada de la Web para cada petición está asociada precisamente con su peso.

Tabla 4.3: Determinación del peso de una petición

Petición	w_1	w_2	w_3	Γ_S
wheat grain tonnes	83,325.88	19,131.12	11,668.86	114,125.86
wheat grain agriculture	83,325.88	19,131.12	1,389.38	103,846.38
wheat grain export	83,325.88	19,131.12	810.95	103,267.95
wheat grain crop	83,325.88	19,131.12	304.58	102,761.58
wheat tonnes corn	83,325.88	11,668.86	304.58	95,299.31
wheat tonnes trade	83,325.88	11,668.86	511.21	95,505.94
wheat tonnes usda	83,325.88	11,668.86	412.61	95,407.35
wheat tonnes washington	83,325.88	11,668.86	255.16	95,249.90
grain tonnes export	19,131.12	11,668.86	810.95	31,610.94
grain agriculture trade	19,131.12	1,389.38	511.21	21,031.71

- ***Búsqueda en la Web***: Usando el conjunto de peticiones para cada categoría como patrón de búsqueda en la Web, se descarga un conjunto de ejemplos no etiquetados para cada categoría. Basado en la observación de qué peticiones más significativas deben recuperar páginas más relevantes a la categoría que pertenecen, el método para descargar información de la Web determina el número de ejemplos que serán descargados para cada petición en función del peso de la misma, esto es, el número de ejemplos descargados está en proporción directa con el valor Γ correspondiente a la petición.

Para un conjunto de M peticiones $\{q_1, \ldots, q_M\}$ para la categoría C, y considerando que se van a descargar N ejemplos por categoría, el número de ejemplos que se van a descargar para una petición q_i en particular está determinado por:

$$\Psi_S(q_i) = \frac{N}{\sum_{k=1}^{M} \Gamma_S(q_k)} \times \Gamma_S(q_i)$$

A manera de ilustrar el uso de esta expresión se considerarán las peticiones de la tabla 4.3, suponiendo que se desea descargar 1,000 ejemplos no etiquetados en total, el número de ejemplos no etiquetados que se descargaría por cada petición se muestra en la tabla 4.4.

Tabla 4.4: Número de ejemplos no etiquetados descargados por petición.

Petición	w_1	w_2	w_3	Γ_S	Ψ_S
wheat grain tonnes	83,325.88	19,131.12	11,668.86	114,125.86	133
wheat grain agriculture	83,325.88	19,131.12	1,389.38	103,846.38	121
wheat grain export	83,325.88	19,131.12	810.95	103,267.95	120
wheat grain crop	83,325.88	19,131.12	304.58	102,761.58	120
wheat tonnes corn	83,325.88	11,668.86	304.58	95,299.31	111
wheat tonnes trade	83,325.88	11,668.86	511.21	95,505.94	111
wheat tonnes usda	83,325.88	11,668.86	412.61	95,407.35	111
wheat tonnes washington	83,325.88	11,668.86	255.16	95,249.9	111
grain tonnes export	19,131.12	11,668.86	810.95	31,610.94	37
grain agriculture trade	19,131.12	1,389.38	511.21	21,031.71	25
				858,106.92	1,000

Como se puede apreciar en la tabla 4.4, el número de ejemplos no etiquetados descargados para cada petición es diferente, en función del peso de la misma. Si se observa, por ejemplo, la primera y la novena petición de la tabla, éstas sólo difieren en una palabra, sin embargo con una se descargan 133 ejemplos no etiquetados, mientras que con la otra se descargaran únicamente 37.

Es importante mencionar que debido a que cada ejemplo descargado corresponde exactamente a una petición en particular, es posible considerar que los ejemplos descargados corresponden a una categoría determinada (la misma categoría de la petición con la cual se descargó el ejemplo no etiquetado). Esto puede ser considerado como la asignación de una categoría a priori del ejemplo descargado, lo cual puede ayudar a mejorar el desempeño del módulo de aprendizaje semi-supervisado. En cierta forma, debido a esta situación, los ejemplos no etiquetados descargados de la Web pueden ser considerados parcialmente etiquetados ya que cada uno de ellos corresponde a una petición y la petición a su vez corresponde a una categoría.

Los ejemplos no etiquetados descargados de la Web son textos cortos. Se eligió descargar snippets en vez de documentos completos debido a que se pensó que la incorporación de poca información, pero altamente discriminativa tendría más impacto en la exactitud del sistema de categorización y en este contexto los snippets resultaron entonces adecuados para probar esta hipótesis. A continuación se muestran dos snippets que fueron descargados utilizando como patrón de búsqueda la petición *grain agriculture trade*.

> Grain Security - Department of Agriculture, Trade & Consumer ... - Trade Practices regulates grain dealers and warehouse keepers with the intent of protecting grain producers from non-payment for grain received by grain ...

> China - AGRICULTURAL TRADE Agricultural trade remains an important component of China's general agricultural modernization effort. China is likely to continue to import grain and ...

Las palabras que acompañan al patrón de búsqueda utilizado, esto es, las palabras de contexto, son las palabras que, una vez seleccionado el ejemplo no etiquetado, enriquecerán al conjunto de entrenamiento y con ello se pretende incrementar la precisión de categorización. Cabe mencionar que los snippets descargados de la Web pasan por una etapa de pre-procesamiento, en la cual son eliminadas las palabras de paro, símbolos y etiquetas especiales. El tamaño promedio de un snippet es de 30 palabras antes de la etapa de pre-procesamiento.

Algunos snippets están formados por varios fragmentos de texto. Este hecho no fue considerado en el presente trabajo en el cual nos hemos limitado a recuperar las palabras contenidas en el snippet, independientemente del fragmento al que pertenezcan, por considerar que están en el contexto del patrón de búsqueda utilizado para la descarga. El incorporar información nueva al conjunto de entrenamiento proveniente de los snippets, es lo que permite mejorar la precisión de la categorización, así como se detallará en los capítulos cinco y seis en los cuales se muestran los resultados experimentales desarrollados en la presente investigación.

Por otro lado, los snippets son seleccionados utilizando el mismo conjunto de entrenamiento entonces aquellos que tengan un mayor parecido con las instancias de entrenamiento original son seleccionados para enriquecer el conjunto de entrenamiento (la selección de los mejores snippets es realizada por el arreglo de métodos de categorización tipo stacking, el cual hace las veces de filtro). Si se agregaran documentos completos, en vez de snippets, se correría el riesgo de provocar un desvió en la temática de la categoría, ya que se puede dar la situación de que la información agregada no etiquetada de la Web sea mayor, en cantidad, que la información etiquetada utilizada como entrenamiento.

En el siguiente apartado se explica el proceso de selección de los ejemplos no etiquetados descargados de la Web el cual, como puede apreciarse, es complementario al mostrado en este apartado.

4.2.2 Aprendizaje semi-supervisado

Como se muestra en la figura 4.1, la segunda parte del método de categorización automática de documentos propuesto consiste en una etapa de aprendizaje semi-supervisado. El objetivo es incrementar la precisión de categorización con un incremento gradual del tamaño del conjunto de entrenamiento original incorporando ejemplos no etiquetados descargados de la Web.

Este proceso está basado en la técnica de self-training, la cual permite el entrenamiento de un arreglo stacking utilizando el conjunto de entrenamiento original y a partir de este, poder seleccionar las mejores instancias no etiquetadas descargadas de la Web. Los ejemplos no etiquetados con la precisión más alta, asignada por el arreglo stacking, son agregados al conjunto de entrenamiento y el proceso se repite.

Como se puede apreciar, el arreglo stacking es el responsable de seleccionar las mejores instancias no etiquetadas que serán incorporadas al conjunto de entrenamiento. A continuación se muestra el algoritmo desarrollado para este fin:

1. Construir un clasificador (C_l) usando un método específico de aprendizaje (l) y un conjunto de entrenamiento (T).

   En este punto es importante comentar que se puede utilizar cualquier método de aprendizaje l[3]. El conjunto de entrenamiento T puede ser muy pequeño, sin embargo, dado el proceso para formar las peticiones descrito en la sección anterior es necesario contar con instancias de entrenamiento para todas las categorías, no importando si el número de ejemplos entre categorías está desbalanceado. Para casos de desbalanceo extremo es necesario aplicar alguna de las técnicas descritas en el capítulo 3, como, por ejemplo, el sub-muestreo. Cabe mencionar que este mismo conjunto de entrenamiento T es utilizado para entrenar el arreglo stacking, el cual se describe más adelante, encargado de seleccionar las mejores instancias no etiquetadas descargadas de la Web.

2. Categorizar los ejemplos no etiquetados descargados de la Web (E) usando el clasificador construido en el punto anterior (C_l). En otras palabras, se estima la categoría para los ejemplos no etiquetados descargados de la Web.

3 En los experimentos llevados a cabo en el presente trabajo, utilizamos naïve Bayes (Eibe et al., 2006) y SVM (Joachims, 2002).

En este paso se asigna el segundo voto de confianza para las instancias descargadas ya que, como se mencionó en la sección anterior, la información descargada lleva asociada una categoría previa en función de la petición utilizada como patrón de búsqueda. De alguna manera ésto puede ser visto como la unión de dos clasificadores C_l y la Web (representado por el conjunto de peticiones).

3. Seleccionar los mejores *m* ejemplos no etiquetados de cada una de las clases $E_m \subseteq E$. En este caso E_m representa el conjunto de snippets seleccionados para incorporarse al conjunto de entrenamiento original y E el conjunto de ejemplos no etiquetados (snippets) descargados de la Web. El proceso de selección de los mejores ejemplos no etiquetados esta basados en las siguientes condiciones:

   - La categoría estimada para un ejemplo no etiquetado corresponde a la categoría de la petición usada para descargarlo.

   - El snippet es uno de los *m* ejemplos seleccionados con la más alta probabilidad de predicción para cada categoría. El arreglo stacking (Bennett et al., 2005) está formado por dos métodos de categorización: naïve Bayes y SVM. Para cada ejemplo no etiquetado estos métodos de categorización le asignan una probabilidad de pertenencia a una categoría. Las instancias no etiquetadas que se les asigna la misma categoría por ambos métodos de categorización son consideradas como ejemplos positivos. Los *m* ejemplos positivos con la más alta probabilidad son seleccionados y son los candidatos a ser incorporados al conjunto de entrenamiento.

4. Combinar los ejemplos seleccionados con el conjunto de entrenamiento original ($T \leftarrow T \cup E_m$) con la finalidad de formar un nuevo conjunto de entrenamiento. Al mismo tiempo los ejemplos seleccionados son eliminados del conjunto de ejemplos no etiquetados descargados de la Web ($E \leftarrow E - E_m$). Cabe mencionar que los *m* mejores ejemplos son para cada categoría y se agregan al conjunto de entrenamiento.

   Iterar σ veces los pasos 1 a 4 o repetir mientras $E_m = \varnothing$. En este caso σ es un valor especificado por el usuario. En la primera iteración los mejores *m* ejemplos seleccionados se agregan al conjunto de entrenamiento original. En la segunda iteración se forma un nuevo conjunto de entrenamiento doblemente enriquecido y así sucesivamente.
   Este proceso se repite hasta que se terminen las instancias no etiquetadas descargadas de la Web categorizadas como positivas por el arreglo stacking.

   Construir un sistema de categorización final utilizando el conjunto de entrenamiento enriquecido. En cada iteración se construye un sistema de categorización que es evaluado con el conjunto de prueba. Este proceso no tiene una condición de paro del todo clara, ya que depende de la cantidad de instancias que se agreguen en cada iteración, del número de instancias categorizadas como positivas por el arreglo stacking y si el conjunto de entrenamiento presenta desbalanceo. En los experimentos llevados a cabo en el presente trabajo se utilizaron dos algoritmos de categorización automática de documentos: Naïve Bayes y SVM, los cuales son ampliamente referenciados en tareas de procesamiento de lenguaje natural y en particular en los sistemas de categorización automática de documentos.

4.3 Aportaciones del método

Respecto a las aproximaciones de categorización textual que utilizan métodos semi-supervisados, el método desarrollado en la presente tesis tiene las siguientes diferencias principales:

- No requiere un conjunto previo de documentos *no* etiquetados, ya que el método propuesto descarga automáticamente ejemplos no etiquetados de la Web. Esta característica es muy importante ya que existen aplicaciones en dominios en los que es muy difícil contar con documentos que puedan ser utilizados para entrenar un sistema de categorización. Por ejemplo, en el área de la identificación de autoría donde sólo existen, generalmente, pocos documentos para cada autor. En este caso el método propuesto no encuentra exactamente esos documentos particulares en la Web, pero localiza fragmentos de texto, que tienen una distribución de palabras similar y entonces pueden ser consideradas como instancias adicionales de entrenamiento.

- El método desarrollado aplica una aproximación basada en self-training para seleccionar las instancias no etiquetadas descargadas de la Web. Para llevar a cabo ésta tarea no sólo se considera la categoría asignada por el sistema de categorización como se hace comúnmente, también se considera una categoría previa que le es asignada al momento de realizar la descarga de la Web, esto es, una categoría a priori asignada al momento de formar las peticiones que serán lanzadas a la Web para descargar ejemplos no etiquetados. El método propuesto utiliza estas dos fuentes complementarias para predecir la categoría de un ejemplo no etiquetado. Este proceso depende del número de instancias etiquetadas que se tengan y, por consecuencia, resulta muy adecuado para realizar trabajos de categorización cuando se tienen muy pocos ejemplos de entrenamiento.

4.4 Conclusiones del capítulo

En este capítulo se presentó la descripción del método semi-supervisado propuesto para llevar a cabo la tarea de categorización automática de documentos. Este método se compone de dos procesos principales: adquisición de corpus de la Web y aprendizaje semi-supervisado.

La etapa correspondiente al proceso de adquisición de corpus es la encargada de descargar información no etiquetada de la Web con la finalidad de incorporar algunos de estos ejemplos al conjunto de entrenamiento. Para llevar a cabo esta tarea primero se construye un conjunto de peticiones, utilizando para ello las palabras con mayor peso de la categoría en cuestión. Estas peticiones son lanzadas a la Web y se descarga información no etiquetada la cual es utilizada por la parte de aprendizaje semi-supervisado.

La etapa correspondiente al proceso de aprendizaje semi-supervisado es la responsable de seleccionar los mejores ejemplos no etiquetados descargados de la Web con la finalidad de incorporarlos al conjunto de entrenamiento original. Para llevar a cabo esta tarea se utiliza un arreglo stacking, el cual es entrenado con el mismo conjunto de entrenamiento y las instancia con la probabilidad más alta asignada por el arreglo stacking son las que se incorporan al conjunto de entrenamiento.

Además, este capítulo presenta las principales aportaciones del método propuesto, entre las que destacan: (i) el no requerir un conjunto previo de información no etiquetada (el método va a la Web y descarga esta información). (ii) Además, considera la incorporación gradual de la mejor información no etiquetada descargada al conjunto de entrenamiento por medio de un proceso iterativo. Esta función es realizada con la finalidad de incrementar la exactitud de categorización.

Capítulo V

Resultados experimentales

En este capítulo se presenta la evaluación experimental del método propuesto en tareas de categorización automática de documentos. Esta evaluación se llevó a cabo en tres diferentes experimentos de clasificación textual, los cuales consideran tanto la categorización temática como la no temática, así como colecciones de documentos en inglés y español. En el primer apartado se describe la parte común a estos experimentos y en los apartados siguientes se presentan los resultados experimentales obtenidos en cada una de las diferentes tareas de categorización evaluadas.

5.1 Configuración general de los experimentos

En este capítulo se presentan los resultados obtenidos en tres diferentes experimentos de categorización de documentos:

(i) Categorización de noticias acerca de desastres naturales en español usando muy pocos ejemplos de entrenamiento;

(ii) Categorización de noticias en inglés de un subconjunto de la colección de Reuters (colección que se caracteriza por tener un alto grado de traslape y desbalanceo);

(iii) Categorización de poemas (no temática), se realizó un experimento acerca de atribución de autoría de un conjunto de poemas correspondientes a poetas contemporáneos mexicanos.

Sin embargo, antes de entrar en la parte de los resultados se menciona la configuración para estos experimentos, es decir, las condiciones comunes que se tomaron en cuenta, las cuales se detallan a continuación.

5.1.1 Búsqueda en la Web

Para llevar a cabo la búsqueda y descarga de información de la Web se utilizó Google como motor de búsqueda. Específicamente se utilizó una API[1] (*Application Programming Interface)* para llevar a cabo esta tarea. Esta interfaz se encuentra disponible por Google para fines de investigación. Esta API permite descargar la información resultante de una búsqueda y genera un archivo para cada ocurrencia o snippet. Se descargaron 1,000 ejemplos no etiquetados adicionales (snippets) para cada categoría en cada experimento.

Como se mencionó en el capítulo anterior, la cantidad de información descargada para cada petición es directamente proporcional al peso de la petición. De esta manera una petición con más peso implica mayor cantidad de información descargada. Lo que se pretende es tener una cantidad mayor de información no etiquetada que contenga las palabras utilizadas para la petición, y de esta manera incrementar la posibilidad de incorporar algunos ejemplos no etiquetados al conjunto de entrenamiento.

1 http://www.google.com/apis

5.1.2 Pre-procesamiento de documentos

El aplicar un pre-procesamiento tanto a los documentos etiquetados como a los no etiquetados nos permite eliminar "ruido" que se pudiera agregar al conjunto de entrenamiento. Esto es, aquella información que únicamente incrementaría la dimensionalidad del espacio de características, el cual es representado por medio de vectores y matrices con el consecuente incremento del coste computacional. El ruido generalmente está formado por información redundante, la cual tiene poco o nulo valor discriminativo y por lo tanto no es útil para llevar a cabo la separación o distinción entre categorías que es lo que buscamos obtener al incorporar nueva información al conjunto de entrenamiento.

En este trabajo se aplican las operaciones tradicionales de pre-procesamiento, las cuales consisten en la eliminación de:

- Etiquetas HTML y XML
- Signos de puntuación
- Números
- Palabras de paro (stop words)

Cabe mencionar que el pre-procesamiento se lleva a cabo tanto en los documentos etiquetados como en los no etiquetados y la finalidad es que ambos conjuntos de datos estén "en igualdad de condiciones" en cuanto a la información que pueden aportar para la construcción del sistema de categorización. Sólo se consideraron palabras (*tokens* alfabéticos), que previamente fueron convertidas a minúsculas.

5.1.3 Algoritmos de aprendizaje

Fueron seleccionados dos algoritmos muy utilizados en el estado del arte en tareas de categorización automática de documentos: naïve Bayes (Lewis 1998; Peng et al., 2004) y las máquinas de vectores de soporte (Joachims 1998; Joachims, 2002). En todos los experimentos llevados a cabo en el presente trabajo fueron considerados como atributos de categorización todas las palabras del conjunto de entrenamiento con una frecuencia mayor que uno. Es decir, las palabras con frecuencia igual a uno fueron eliminadas, al considerar que su aporte para distinguir una categoría de otra es nulo.

No se utilizó ganancia de información para seleccionar las características de categorización debido a que el número de características se reduce significativamente provocando que al construir el sistema de categorización con muy pocos atributos de entrenamiento este se confunda al decidir sobre la categoría que asignará a una instancia de prueba. Sin embargo, si se utiliza la ganancia de información para darle peso a las palabras a la hora de formar las peticiones. Para llevar a cabo la selección de las mejores instancias no etiquetadas, es decir, aquellas que serán incorporadas al conjunto de entrenamiento, se hace uso del arreglo stacking, el cual está formado por dos algoritmos de categorización: uno basado en naïve Bayes y otro basado en SVM. En el siguiente apartado se presentan las medidas de evaluación utilizadas para evaluar la efectividad del método de categorización implementado.

5.1.4 Medidas de evaluación

La efectividad del método fue medida por la exactitud de categorización, la cual indica el porcentaje de documentos del conjunto de prueba que fueron categorizados correctamente. En todos los casos se llevó a cabo la medición de la significancia estadística. La prueba de significancia estadística utilizada consiste en la prueba-*t* o *t*-test (*paired students t-test*), la cual se aplicó con un valor de confianza de 0.005 (Smucker et al., 2007).

Así como se mencionó anteriormente, básicamente existen dos maneras de llevar a cabo la evaluación de un sistema de categorización de documentos: tener conjuntos de entrenamiento y prueba separados o llevar a cabo una validación cruzada, comúnmente un 10-*fold cross validation*, donde el conjunto de entrenamiento es partido en 10 subconjuntos de los cuales uno a la vez es utilizado para probar el sistema obteniendo al final el promedio de exactitud de todas las partes (Salzberg, 1999). En tareas de categorización automática de documentos el mejor escenario es el primero: el de tener conjuntos de entrenamiento y prueba separados, debido a que el conjunto de prueba no es visto nunca por el conjunto de entrenamiento. Ésta fue la evaluación que utilizamos en este trabajo de tesis.

Adicional a estas medidas que podríamos llamar tradicionales, y con la finalidad de comprender mejor los resultados obtenidos, se realizó un análisis estadístico de los mismos. El propósito de este análisis es complementar la explicación respecto del desempeño de los sistemas de categorización basados en naïve Bayes y SVM utilizados en el presente trabajo. Es importante señalar que naïve Bayes es un categorizador probabilístico que aplica el teorema de Bayes para hacer la predicción de la categoría asignada a los elementos del conjunto de prueba. Esta asignación se lleva a cabo suponiendo (ingenuamente) que existe una independencia entre las categorías de los elementos que se van a categorizar. Desde este punto de vista, es conveniente utilizar una medida estadística que tome en cuenta la relación entre las palabras que conforman cada uno de los textos (las características utilizadas). En particular se utilizó la medida llamada *SLMB*[2] (*Supervised Language Modeling Based*) (Pinto, 2008).

Esta medida, *SLMB*, utiliza un conjunto de modelos de lenguaje (basados en bi-gramas y tri-gramas) para calcular la entropía entre las distintas categorías en las que se puede ubicar un documento. Formalmente tenemos que dado un corpus *D*, con un *gold standard* compuesto de *k* categorías $C = \{C_1, C_2,\ldots,C_k\}$, la medida *SLMB* se define de la siguiente manera:

$$SMLB(D) = \sqrt{\frac{1}{k}\sum_{i=1}^{k}(Perplexity(C_i \mid \overline{C_i^*}) - \mu(Perplexity(C)))^2}$$

2 http://nlp.dsic.upv.es:8080/watermaker

Donde $\overline{C_i^*}$ indica el modelo del lenguaje obtenido al utilizar todas las categorías excepto C_i. La perplejidad del modelo de lenguaje de la categoría C_i, con respecto al modelo de lenguaje $\overline{C_i^*}$, está representada por $Perplexity(C_i \mid \overline{C_i^*})$. La media de la perplejidad a través de las diferentes categorías en las que puede ser ubicado un documento se calcula de la siguiente manera, donde *k* es el número de categorías.

$$\mu(Perplexity(C)) = \frac{\sum_{i=1}^{k} Perplexity(C_i \left| \overline{C_i^*} \right.)}{k}$$

Esta medida intenta capturar la complejidad de distinguir las categorías a través de modelos de lenguaje. En los experimentos realizados en el presente trabajo de investigación se construyeron los modelos de lenguaje a partir de un vocabulario cerrado al utilizar el conjunto de entrenamiento original. Podríamos considerar modelos de lenguaje abierto cuando se utilizaron los conjuntos de entrenamiento enriquecidos incorporando información no etiquetada de la Web. A la hora de calcular el modelo se utilizaron los conjuntos pre-procesados, esto es, las palabras de paro (palabras vacías), símbolos y etiquetas especiales, fueron removidas de los documentos. Como se podrá ver en las tablas que muestran los resultados de esta medida, el valor de esta medida se incrementa con las iteraciones del método semi-supervisado. Esto se debe a que las palabras distintas entre los documentos incrementan este valor y es precisamente lo que ocurre al incorporar información no etiquetada de la Web al conjunto de entrenamiento: se incorpora nuevo vocabulario. También se llevó a cabo la medición tanto de las palabras distintas en el corpus así como del tamaño del vocabulario o total de palabras en el mismo. Esta medición es interesante, ya que permite ver el incremento del vocabulario entre las iteraciones realizadas y se puede dar una idea muy clara de la cantidad de información nueva que se incorpora al conjunto de entrenamiento. Como medida de evaluación, aunque más visual, también se emplearon gráficas de similitud entre las diferentes categorías. La finalidad de estas gráficas es que permitan entender mejor los resultados obtenidos principalmente en cuanto al beneficio que representa la incorporación de información nueva al conjunto de entrenamiento y el impacto que ésta tiene en la mejora de la exactitud del sistema de categorización. En cada experimento se detalla la interpretación de cada gráfica de similitud.

5.2 Categorización de noticias sobre desastres naturales

En esta sección se presentan los resultados obtenidos sobre una colección de noticias en español acerca de desastres naturales. Este experimento se caracteriza por utilizar muy pocos ejemplos de entrenamiento. Específicamente se llevaron a cabo experimentos con 1, 2, 5 y 10 ejemplos de entrenamiento y en todos los casos se utilizó un conjunto homogéneo de prueba. Los detalles de los experimentos se presentan a continuación.

5.2.1 Objetivo del experimento

El primer experimento realizado se enfocó en la categorización de noticias en español acerca de desastres naturales. El objetivo fue evaluar el método propuesto en un escenario típico en una lengua distinta al inglés y considerando muy pocos ejemplos de entrenamiento para un dominio específico de aplicación.

Adicional a este objetivo general, este experimento también considera la evaluación de algunos de los componentes del método propuesto, en particular, la evaluación del sistema de filtrado basado en la Web utilizado para seleccionar los mejores ejemplos no etiquetados que serán incorporados al conjunto de entrenamiento. A continuación se presenta una descripción del corpus utilizado y los resultados obtenidos para este experimento.

5.2.2 Descripción del corpus

Para este experimento, el corpus utilizado fue un conjunto de noticias en español acerca de desastres naturales. Este corpus fue colectado de los periódicos mexicanos Reforma[3] y El Universal[4], en su versión electrónica.

El corpus se compone de 240 documentos agrupados en cuatro categorías distintas: Incendio Forestal (C1), Huracanes (C2), Inundaciones (C3) y Terremotos (C4). Cabe mencionar que este corpus se encuentra disponible para investigadores interesados[5]. Con la finalidad de mostrar la forma y estructura, así como resaltar la dificultad de esta tarea, a continuación se presenta una noticia de este corpus, en este caso corresponde a una noticia de la categoría inundación del conjunto de prueba.

El Universal

Decretan alerta en Honduras por inundaciones

Tegucigalpa.- El gobierno hondureño decretó hoy un estado de alerta preventivo en la costa atlántica del país, luego que torrenciales aguaceros causados por un frente frío inundaron extensos territorios de la región. Las lluvias asimismo se han extendido a todo el país. Ante esa situación, las autoridades cerraron los cuatro aeropuertos internacionales, tras cancelar brevemente los vuelos en Tegucigalpa, en la región central, y San Pedro Sula, La Ceiba e Islas de la Bahía, al norte de Honduras, informó Ap. "La situación se agravará porque esperamos más lluvias en los próximos dos días", dijo a la AP el vocero de la Comisión Permanente de Contingencia (COPECO), Carlos Gonzáles. Las calles principales de La Ceiba y Puerto Cortés están inundadas desde el martes. Las lluvias han provocado en ambas ciudades y sus alrededores pérdidas aún no calculadas, según Gonzáles.

Las aguas también han afectado numerosos negocios, viviendas y carreteras del litoral atlántico hondureño, tras alcanzar una altura de hasta dos metros en muchas zonas. Las autoridades hacen esfuerzos por sacar gran cantidad de automóviles de las calles inundadas, donde las aguas han formado lagunas por la intensidad de las lluvias. Allí permanecen desde el martes. Miembros de la COPECO y socorristas de la Cruz Roja han evacuado a por lo menos 800 personas en dos días.

3 http://www.reforma.com
4 http://www.eluniversal.com.mx
5 http://ccc.inaoep.mx/~mmontesg/resources/Desastres.sgm

Como se puede observar, para un anotador humano es fácil ubicar esta noticia dentro de la categoría inundación, pero para un sistema de categorización automática no es una tarea trivial, ya que por ejemplo esta noticia bien podría caber dentro de la categoría de huracanes, los cuales comúnmente ocasionan inundaciones.

Para llevar a cabo la evaluación experimental, el corpus fue organizado de la siguiente manera: cuatro diferentes conjuntos de entrenamiento formados por 1, 2, 5 y 10 ejemplos por categoría respectivamente y un conjunto fijo de prueba de 200 ejemplos (50 documentos por categoría). En el siguiente apartado se presentan los resultados obtenidos con la evaluación de este corpus.

5.2.3 Resultados

En este apartado se presentan los resultados obtenidos, tanto en el experimento de referencia (baseline) como en el conjunto de entrenamiento enriquecido después de haber aplicado el método semi-supervisado basado en la Web.

5.2.3.1 Resultados de referencia

Este resultado corresponde a la aplicación directa del método de categorización seleccionado (*NB* o *SVM*) sobre el conjunto de prueba utilizando los diferentes conjuntos de entrenamiento. En la tabla 5.1 se muestran estos resultados (porcentajes de exactitud) para las cuatro condiciones distintas de entrenamiento que se utilizaron.

Tabla 5.1: Resultado de referencia para la colección de desastres

Número de ejemplos de entrenamiento	NB	SVM
1	51.7	50.0
2	56.7	58.3
5	80.4	77.1
10	77.1	80.4

En estos resultados podemos observar como las aproximaciones tradicionales de categorización proporcionan, en general, un resultado pobre cuando se utilizan muy pocos ejemplos de entrenamiento. Sin embargo, es de resaltar que, con sólo 5 ejemplos de entrenamiento por categoría se obtiene un destacable 80% de exactitud. También se puede observar que se mantiene una relación directa entre el número de instancias de entrenamiento y la exactitud de categorización obtenida, lo cual es lógico ya que se tienen más instancias para aprender.

El comportamiento de los métodos de categorización es similar, y se puede observar que entre más pequeño es el conjunto de entrenamiento, más sensible es a la información que se añade. Sin embargo, debido a que son muy pocas instancias de entrenamiento, es probable que se agreguen algunas instancias "confusas" dado que el sistema no aprendió lo suficiente y tal vez a esto se deba el bajo rendimiento. También se puede observar en el caso de naïve Bayes un decremento en la exactitud de categorización cuando se utilizan diez ejemplos de entrenamiento respecto de cuando se utilizan cinco. Esto se puede deber a que el número de ejemplos es muy pequeño para que el sistema pueda aprender y tiene relación también con el tamaño del ejemplo, vocabulario y calidad del mismo.

5.2.3.2 Resultados obtenidos al aplicar el método propuesto

Con la finalidad de evaluar el método propuesto, se diseñaron dos experimentos en los cuales el valor del atributo *m* del algoritmo descrito en el capítulo anterior, fue modificado como se describe a continuación:

1. En cada iteración se agregó solamente un ejemplo no etiquetado a cada categoría del conjunto de entrenamiento; es decir *m=1*.

2. En cada iteración se agregan al conjunto de entrenamiento un número de ejemplos no etiquetados igual al número de ejemplos de entrenamiento en la colección original, esto es, $m=|T|$.

En la tabla 5.2 se muestran los resultados obtenidos para estos experimentos utilizando el clasificador naïve Bayes. En esta tabla, algunas de las cantidades tienen un "*" después del valor de la exactitud de categorización. Esto indica que el resultado obtenido respecto al resultado de referencia es estadísticamente significativo.

Tabla 5.2: Exactitud con NB (m=1 y m=$|T|$), colección desastres

Ejemplos de Entrenamiento	Resultados de referencia	Valor *m*	Método propuesto		
			1ra iteración	2da iteración	3ra iteración
1	51.7	*m*=1	78.3*	77.3*	76.0*
2	56.7		70.0*	86.0*	86.1*
5	80.4		82.2	85.1	92.1*
10	77.1		83.1	87.2*	91.3*
1	51.7	*m*=|*T*|	78.3*	77.3*	76.0*
2	56.7		86.5*	87.6*	86.5*
5	80.4		97.0*	96.5*	95.6*
10	77.1		97.2*	97.5*	96.5*

Como se puede apreciar en la tabla 5.2, los mejores resultados se obtienen cuando se utiliza $m=|T|$. Esto se debe a que en un número menor de iteraciones se tiene una cantidad de información mayor. Por ejemplo, para el caso de diez ejemplos de entrenamiento, en la segunda iteración ya se tendrían treinta documentos (diez originales y veinte no etiquetados que se agregaron) logrando un destacable 97.5% de exactitud.

Sólo se llevaron a cabo tres iteraciones ya que, como se describió en el objetivo del experimento, la idea era ver el efecto de incorporar información no etiquetada, proveniente en este caso de la Web, en la exactitud de categorización. En particular cuando $m=1$ se puede intuir que, como se agrega muy poca información en cada iteración, sería de esperar una mejora continua, aunque lenta, en la exactitud de categorización ya que se tendrían que llevar a cabo T iteraciones para igualar la exactitud obtenida con $m=|T|$.

En la tabla 5.3 se muestran los resultados obtenidos para el mismo experimento, pero ahora utilizando como algoritmo de categorización base SVM. De la misma manera que en el caso anterior, el "*" indica significancia estadística.

Tabla 5.3: Exactitud con SVM ($m=1$ y $m=|T|$), colección desastres

			Método propuesto		
Ejemplos de Entrenamiento	**Resultados de referencia**	**Valor m**	**1ra iteración**	**2da iteración**	**3ra iteración**
1	50.0	m=1	49.1	51.0	55.3
2	58.3		62.3	68.1*	67.0
5	77.1		76.4	80.1	87.0*
10	80.4		82.1	85.2	90.1*
1	50.0	m=|T|	49.1	51.0	55.3
2	58.3		68.2*	74.0*	74.5*
5	77.1		93.5*	92.5*	96.0*
10	80.4		96.5*	96.1*	95.1*

En estos resultados se puede observar que se obtienen mejores resultados utilizando el enriquecimiento del conjunto de entrenamiento original con $m=|T|$. El comportamiento es similar al obtenido en el caso anterior (véase la tabla 5.2) ya que entre más ejemplos de entrenamiento se tengan en el conjunto original, mejores resultados se obtienen, debido a que se puede llevar a cabo una mejor selección de la información no etiquetada. Se puede notar también que los mejores snippets son seleccionados en las primeras iteraciones; por eso, por ejemplo, en la iteración número tres se ve un ligero descenso en la exactitud obtenida para el caso de diez ejemplos de entrenamiento en el conjunto original.

En general podemos concluir que los resultados experimentales obtenidos son muy satisfactorios ya que, como se puede observar, claramente se mejoran los resultados de referencia usando cualquiera de los sistemas de categorización. En particular, con $m=|T|$ podemos observar una mejora en la exactitud de más del 25%. Este resultado es muy relevante ya que agregando sólo pocos ejemplos (los cuales son documentos muy pequeños) al conjunto de entrenamiento original se puede tener una mejora significativa en los resultados de categorización. Estos resultados también nos permiten observar que en todos los casos se obtienen mejores resultados con naïve Bayes que con SVM. Además, se aprecia que desde la primera iteración la exactitud obtenida con naïve Bayes es significativamente mayor que la obtenida con SVM. Esto se debe a la manera cómo funcionan los sistemas de categorización: al ser muy poca información los vectores de soporte están incompletos para poder predecir la categoría de un ejemplo no etiquetado, mientras que Bayes asigna la categoría a una instancia por medios probabilísticos.

Asi como se pudo observar en las Tablas 5.2 y 5.3 los mejores resultados se obtienen para un conjunto de entrenamiento formado por diez ejemplos por categoría, lo cual confirma nuestra idea intuitiva, ya que al contar con un número mayor de ejemplos, se cuenta con más infomación para que el sistema pueda aprender.

Con la finalidad de comprender mejor el desempeño de ambos sistemas de categorización, naïve Bayes y SVM, se llevó a cabo un análisis estadístico de los corpus utilizados. En particular se llevó a cabo la obtención de la medida llamada *SLMB,* descrita en el apartado 5.1.4. Así como una comparación entre el vocabulario original y el vocabulario que se tiene en cada iteración para observar claramente de cómo es su crecimiento con relación a la nueva información no etiquetada, proveniente de la Web, que se incorpora al conjunto de entrenamiento. Los resultados se resumen en la tabla 5.4.

Tabla 5.4: SLMB y vocabulario de la colección de desastres

	Original			Enriquecido		
	SLMB	**PD**	**TP**	**SLMB**	**PD**	**TP**
1	1.12	448	629	50.07	595	871
2	109.52	770	1163	134.09	960	1601
5	790	1564	2914	141.85	1959	4032
10	93.20	2776	6193	190.42	3368	8463

En la tabla se muestran los resultados obtenidos al aplicar la medida SLMB en los conjuntos de entrenamiento original y enriquecido. Estos conjuntos de entrenamiento son independientes, motivo por el cual no se ve una relación clara con respecto a la medida SLMB y los diferentes conjuntos de entrenamiento, aunado a que son muy pocos archivos en el conjunto de entrenamiento original para poder concluir algo al respecto. Sin embargo, se puede observar que la incorporación de nuevo vocabulario al comparar el número de palabras distintas (PD) y el total de palabras (TP) entre los conjuntos original y enriquecido. Aquí podemos ver que la incorporación de información no etiquetada al conjunto de entrenamiento realmente provee vocabulario nuevo, lo que hace que los vectores de categorización, así como los modelos de lenguaje también se modifiquen logrando con ello un incremento en la exactitud de categorización. Dados los resultados obtenidos, podemos observar que la información que se incorpora al conjunto de entrenamiento, no solo contiene nuevo vocabulario, sino que además este nuevo vocabulario permite incrementar las diferencias entre las categorías. Un valor grande de SLMB significa una mayor diferencia entre las categorías. Al comparar el conjunto de entrenamiento original con el conjunto de entrenamiento enriquecido podemos observar que en todos los casos aumento.

Con la finalidad de tener una interpretación visual del impacto que tiene la información no etiquetada que se incorpora al conjunto de entrenamiento, se realizaron gráficas de similitud. Estas gráficas corresponden a la matriz generada al categorizar el conjunto de prueba utilizando como atributos de categorización los correspondientes al conjunto de entrenamiento original y enriquecido formado por 10 ejemplos de entrenamiento originales que es donde se obtienen los mejores resultados. El conjunto de prueba está formado por 200 archivos, correspondientes a cuatro categorías distintas. Para llevar a cabo estas gráficas se forma una matriz de 200x200 y cada archivo es comparado contra todos los demás, incluido el mismo. De esta manera si dos archivos son iguales el valor de similitud será 1 (caso de la diagonal principal de la matriz) y en caso de no ser nada similares el valor de similitud será cero. En las gráficas representamos el valor de 1 con negro y de 0 con blanco. Cabe mencionar que para poder distinguir mejor el resultado se utilizó un umbral de similitud. En este caso solo los valores que superan este umbral son mostrados en la gráfica.

En la figura 5.1 se muestra la gráfica de similitud para el conjunto de prueba. Se utilizaron como atributos de categorización, en este caso, los correspondientes al conjunto de entrenamiento de diez ejemplos etiquetados por categoría. El umbral de similitud utilizado fue de 0.3. Como se puede apreciar en la gráfica las categorías en las que más se confunde el sistema de categorización son *Inundación* y *Huracán*, lo cual es hasta cierto punto lógico, ya que es común que los huracanes generen inundaciones. Sin embargo, y a pesar del número reducido de instancias de entrenamiento, se puede ver como el conjunto de prueba es separado en cuatro grupos bien definidos que corresponden a los archivos que tienen mayor similitud entre ellos y que nos permite ubicar cada grupo como una categoría. También podemos ver que en las categorías que existe menos confusión son en *Sismo* y *Forestal*. Esto se puede deber a que manejan vocabularios con poco traslape entre ellos, lo cual permite separar mejor las categorías.

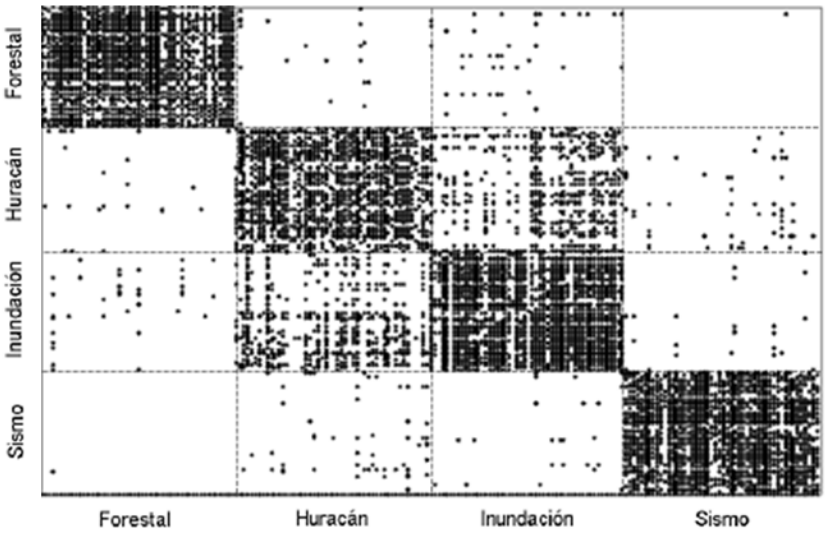

Figura 5.1: Gráfica de similitud del conjunto de prueba, colección de desastres

En la figura 5.2 se muestra la gráfica de similitud para este mismo corpus de prueba, pero utilizando como atributos de categorización los correspondientes al conjunto de entrenamiento formado por diez ejemplos de entrenamiento en la tercera iteración (40 archivos, 10 originales y 30 incorporados, 10 en cada iteración). La idea de mostrar esta gráfica es ver cómo se logra disminuir la confusión que existe en el conjunto original al incrementar el número de instancias etiquetadas de entrenamiento. En este caso también se utilizó un umbral de similitud igual a 0.3.

Como se puede apreciar en las figuras 5.1 y 5.2, las categorías están bien definidas en el conjunto de prueba. Sin embargo, se nota una densidad mayor en la figura 5.2 correspondiente a la categorización del conjunto de prueba utilizando como atributos de categorización el conjunto enriquecido con 10 ejemplos de entrenamiento original y la tercer iteración. Esto quiere decir que la información no etiquetada incorporada, proveniente de la Web, en realidad ayuda a distinguir entre las categorías, a pesar de ser muy pocos los ejemplos de entrenamiento proporcionados originalmente.

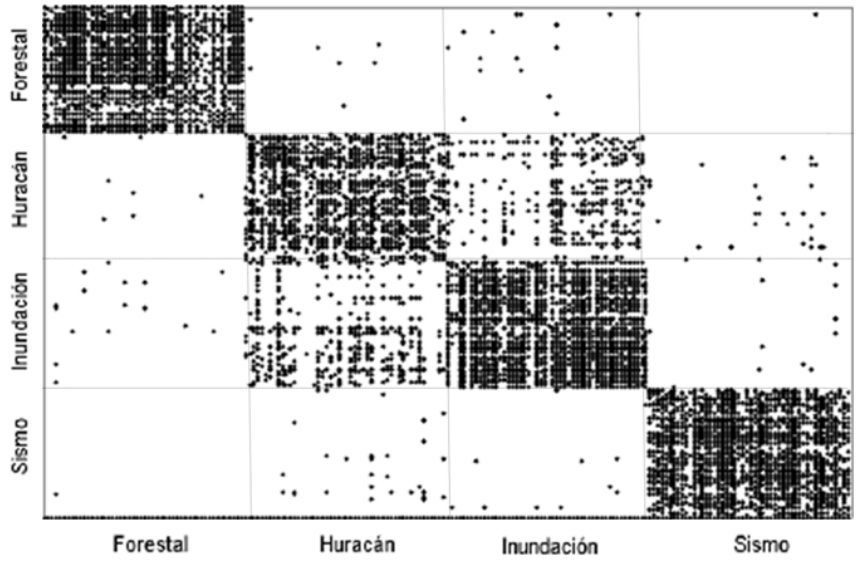

Figura 5.2: Grafica de similitud del conjunto de entrenamiento enriquecido, colección de desastres

Adicionalmente al punto anterior, llevamos a cabo otro experimento con la finalidad de evaluar el método semi-supervisado basado en la Web para la selección de los mejores *m* ejemplos no etiquetados (ver el paso 3 del algoritmo mostrado en el capítulo anterior). Con este experimento se pretende dar respuesta a las siguientes preguntas de investigación:

- ¿Qué tan efectivo resulta el proceso de selección de snippets?
- ¿Qué tan valiosa es la categoría previa asignada por la Web?

Para responder a estas preguntas, se llevó a cabo un experimento omitiendo la etapa de selección de snippets formada por el arreglo stacking. Es decir, se llevó a cabo la selección de las instancias no etiquetadas de la manera tradicional, esto es, no tomando en cuenta la categoría previa asignada por la Web, sino considerando todos los snippets descargados, independientemente de la categoría de la petición, como parte de una bolsa de snippets y de ahí se fueron seleccionando. La tabla 5.5 resume los resultados de este experimento al usar el sistema de categorización basado en naïve Bayes.

Tabla 5.5: Exactitud sin filtro de selección de snippets usando NB

Ejemplos de entrenamiento	Resultados de referencia		Valor m	Método propuesto sin selección de snippets (Iteraciones)		
	NB	SVM		1ra	2da	3ra
1	51.7	50.0	m=1	59.9	61.5	54.0
2	56.7	58.3		76.5	79.5	83.5
5	80.4	77.1		88.0	81.5	80.0
10	77.1	80.4		90.5	90.0	88.0
1	51.7	50.0	m=|T|	59.5	61.5	54.0
2	56.7	58.3		76.0*	74.0*	77.5*
5	80.4	77.1		83.1	83.0	80.5
10	77.1	80.4		86.0*	77.5	84.5

Los valores de exactitud que se muestran en la tabla 5.5 son menores que los que se mostraron en la tabla 5.2. Además, un número menor de resultados mostraron significancia estadística. En la tabla 5.6 se muestran los resultados obtenidos en este experimento al usan SVM.

Tabla 5.6: Exactitud sin filtro de selección de snippets usando SVM

Ejemplos de entrenamiento	Resultados de referencia		Valor m	Método propuesto sin selección de snippets (iteraciones)		
	NB	SVM		1ra	2da	3ra
1	51.7	50.0	m=1	46.5	44.0	34.5
2	56.7	58.3		66.5	62.5	66.5
5	80.4	77.1		84.5	79.0	79.0
10	77.1	80.4		91.5*	86.0	79.5
1	51.7	50.0	m=|T|	46.5	44.0	34.5
2	56.7	58.3		56.5	52.0	61.5
5	80.4	77.1		61.0	66.5	66.5
10	77.1	80.4		67.5	71.0	65.0

Estos resultados son muy interesantes ya que, si los comparamos con los obtenidos en las tablas 5.2 y 5.3, podemos observar que:

- El método propuesto basado en la Web filtra y etiqueta los ejemplos no etiquetados permitiendo seleccionar las mejores instancias. Aunque se puede apreciar una mejora con respecto a los resultados obtenidos en el resultado de referencia, los valores de exactitud obtenidos en este último experimento son menores que los obtenidos en el experimento previo. Además, sólo unos pocos de estos resultados muestran una significancia estadística.

- En este experimento, en la mayoría de los casos, el mejor resultado fue obtenido en la primera iteración. Este hecho nos muestra que el módulo de filtrado juega un rol muy importante en el tratamiento de los ejemplos no etiquetados con un valor de predicción medio y bajo, los cuales pueden ser seleccionados en una iteración posterior. Por esta razón en el experimento previo (particularmente en el caso de SVM) los mejores resultados fueron generalmente obtenidos en la tercera iteración.

5.3 Categorización de noticias del corpus de Reuters

En este apartado se presentan los resultados obtenidos al aplicar el método propuesto a la categorización de noticias en inglés correspondientes a la colección de Reuters. Cabe mencionar que esta es una colección estándar muy utilizada en tareas de categorización textual. A continuación se presenta el objetivo del experimento así como la descripción del corpus utilizado. Esto permitirá ubicar la complejidad y características especiales de categorizar esta colección.

5.3.1 Objetivo del experimento

El propósito de este experimento tiene una doble finalidad. Por un lado, validar la independencia del lenguaje del método propuesto, y por otro lado, evaluar su desempeño en una colección de documentos grande. Con la finalidad de cumplir este objetivo se consideró la categorización de documentos de un subconjunto de la colección Reuters, la cual consiste de más de 10,000 documentos en inglés para 10 diferentes categorías.

5.3.2 Descripción del corpus

Para este experimento, fue seleccionado el subconjunto de las 10 categorías más utilizadas en experimentos de categorización de textos de la colección Reuters-21578. En particular se consideró la distribución ModApte[6], la cual consiste en 7,206 instancias de entrenamiento (documentos publicados antes de 04/07/87) y 3,220 instancias de prueba (documentos publicados después del 04/07/87) por parte de la agencia de noticias Reuters.

6 http://www.daviddlewis.com/resources/testcoleections/Reuters21578

Esta distribución contiene las diez categorías con más ejemplos y, por lo tanto, las más utilizadas en experimentos de categorización automática de documentos (Lewis et al., 1994; Sebastiani, 2006). En la tabla 5.7 se muestra el número de documentos de entrenamiento y prueba para cada una de las categorías que conforman esta distribución.

Tabla 5.7: Distribución ModApte de Reuters

Categoría	Conjunto de entrenamiento	Conjunto de prueba
acq	1650	798
corn	182	71
crude	391	243
earn	2877	1110
grain	434	194
interest	354	159
money-fx	539	262
ship	198	107
trade	369	182
wheat	212	94
Total	7206	3220

A continuación se presentan algunos datos relacionados con el corpus utilizado y los resultados obtenidos en este experimento, tanto de referencia como utilizando el método propuesto.

5.3.3 Resultados

En este apartado se presentan los resultados obtenidos, tanto en el experimento de referencia (baseline) como en el conjunto de entrenamiento enriquecido después de haber aplicado el método semi-supervisado basado en la Web. Cabe mencionar que en este caso la tarea de categorización es más difícil que el experimento anterior debido a que:

- Es mayor el número de categorías (en este caso diez)
- El número de instancias de entrenamiento y prueba es significativamente mayor
- Existe un alto grado de traslape entre las categorías
- Existe desbalanceo crítico entre las categorías

5.3.3.1 Resultados de referencia

Como se puede apreciar en la tabla 5.7, la colección considerada para este experimento presenta un alto grado de desbalanceo entre las categorías. Adicional a esto, las categorías presentan también un alto grado de traslape lo que hace aun más difícil la tarea de categorización. El traslape es evidente en la tabla 5.8. En esta tabla se muestran las palabras relevantes para la categoría *wheat*, siguiendo el proceso descrito en el capítulo 4. En ella se pueden observar las palabras *corn*, *trade* y *grain*, las cuales corresponden a los nombres de otras categorías de la distribución ModApte. Además, como se puede apreciar, estas palabras hacen referencia a un contexto similar, por lo que varios de los ejemplos pertenecerán a más de una de estas categorías; de hecho en un análisis de estas categorías que se llevó a cabo identificando aquellos ejemplos de entrenamiento que sólo pertenecen a una y sólo una categoría, se encontró que para las categorías *corn* y *wheat* no había un solo ejemplo que perteneciera únicamente a estas categorías (Guzmán et al., 2007b). Este resultado evidencia el traslapa entre las categorías

Tabla 5.8: Palabras relevantes para la categoría *wheat*

Palabra	f	IG	f*IG
wheat	418	199.34	83325.87
grain	164	116.65	19131.12
tonnes	210	55.56	11668.86
corn	89	39.51	3516.92
agriculture	75	18.52	1389.37
trade	42	12.17	511.20
export	76	10.67	810.95
usda	54	7.64	412.61
crop	51	5.97	304.57
washington	44	5.79	255.16

Considerando que nuestro método no es inmune a este problema, y aprovechando que es especial para trabajar con muy pocos ejemplos de entrenamiento, se decidió configurar el experimento como se indica a continuación:

- Se redujo el tamaño del conjunto de entrenamiento, dejando a todas las categorías con el mismo número de instancias. En otras palabras, el conjunto de entrenamiento fue balanceado aplicando el método de sub-muestreo (*under-sampling)*. La finalidad de este proceso es tener un conjunto de entrenamiento pequeño y balanceado.

- Se aplicó el método de aprendizaje semi-supervisado basado en la Web con la finalidad de ir incorporando nueva información que nos permita distinguir entre las categorías. Ahora el conjunto de entrenamiento va creciendo en la misma proporción, ya que se agrega el mismo número de instancias no etiquetadas a cada categoría, evitando con esto que una categoría se "coma" a otra por tener una gran diferencia en el tamaño del vocabulario.

Sin embargo, al llevar a cabo la configuración del experimento tomando en cuenta estos puntos, en particular con el sub-muestreo, surge la siguiente pregunta:

- ¿Cuántos ejemplos eliminar?, es decir, ¿Cuántos ejemplos de entrenamiento es conveniente dejar en cada categoría?

Para dar respuesta a estas preguntas se llevó a cabo la obtención de la exactitud de categorización utilizando un número diferente de instancias de entrenamiento por categoría. La tabla 5.9 muestra los resultados obtenidos utilizando diferentes niveles de reducción de datos y la exactitud obtenida en cada caso. Cabe mencionar que en todos los casos se utilizó para la evaluación el conjunto de prueba proporcionado, esto es los 3,220 documentos.

Tabla 5.9: Exactitud para diferentes conjuntos de entrenamiento, Reuters

Ejemplos de entrenamiento por categorías	Exactitud
10	58.6
20	73.7
30	77.3
40	79.3
50	81.8
80	82.8
100	84.1
Resultado de referencia	84.7

Es importante notar que usando sólo 100 ejemplos de entrenamiento por categoría se obtiene un resultado muy cercano al obtenido en el resultado de referencia, el cual corresponde al uso de todo el conjunto de entrenamiento (7,206 instancias de entrenamiento). Se puede observar, además, que existe una relación directa entre el número de ejemplos de entrenamiento por categoría y la exactitud de categorización. Éste hecho se observa mejor en los primeros conjuntos, que es cuando se va incorporando una cantidad mayor de vocabulario nuevo al conjunto de entrenamiento, al incrementar el número de ejemplos de entrenamiento por categoría.

5.3.3.2 Resultados obtenidos al aplicar el método propuesto

Con la finalidad de evaluar el impacto del método de categorización usando aprendizaje semi-supervisado y basado en la Web, se diseñaron dos experimentos utilizando solamente diez y cien instancias de entrenamiento por categoría respectivamente. Se eligieron los conjuntos con diez y cien instancias de entrenamiento porque en el caso de diez nos permitiría evaluar el desempeño del método cuando se tienen muy pocos ejemplos de entrenamiento y un alto grado de traslape entre las categorías, mientras que el conjunto de cien fue por su cercanía con el resultado de referencia.

En ambos experimentos se llevaron a cabo diez iteraciones agregando en cada una los mejores diez snippets a cada categoría. La tabla 5.10 muestra la exactitud obtenida en estos experimentos.

Tabla 5.10: Exactitud usando 10 y 100 instancias de entrenamiento, Reuters

	Exactitud	
	Usando 10 ejemplos etiquetados por categoría	**Usando 100 ejemplos etiquetados por categoría**
Valor inicial	58.6	84.1
Iteración 1	66.9*	84.6
Iteración 2	68.7*	84.7
Iteración 3	69.6*	84.8
Iteración 4	70.3*	86.6*
Iteración 5	70.6*	86.8*
Iteración 6	68.6*	86.9*
Iteración 7	69.0*	86.7*
Iteración 8	69.0*	86.7*
Iteración 9	68.5*	86.7*
Iteración 10	68.7*	86.7*

Las cantidades que tienen un asterisco "*", representan los casos en los cuales el método propuesto obtiene una significancia estadística sobre el valor inicial de exactitud (resultado de referencia). La intención en ambos casos fue la de ir agregando información no etiquetada de manera gradual a todas las categorías. De esta manera el corpus de entrenamiento va creciendo de forma balanceada y nos permite ver el efecto que tiene la información no etiquetada en cada iteración. Así como se puede inferir, las peticiones construidas por el conjunto de entrenamiento que contiene cien ejemplos son más pertinentes a la categoría, por lo que los resultados obtenidos son mejores. Es importante mencionar que la descarga de información se realiza una sola vez y después, por medio del proceso iterativo, se incorporan los mejores snippets al conjunto de entrenamiento. En la tabla es posible observar el impacto del método semi-supervisado propuesto. Por ejemplo, cuando usamos sólo diez ejemplos de entrenamiento por categoría, el método obtiene un notable 12% de incremento en la exactitud de (58.6 % a 70.6%).

Sin embargo, dada la complejidad de la colección de prueba (que contiene traslape entre las categorías, por ejemplo en: *grain*, *corn* y *wheat*) es necesario comenzar con más ejemplos de entrenamiento si se desea superar el resultado de referencia obtenido utilizando el conjunto de entrenamiento completo.

En el caso del segundo experimento (en el cual usamos cien ejemplos de entrenamiento por categoría) el incremento en la exactitud no fue tan alto como en el primero. En este caso solamente se obtiene un incremento del 84% al 86.9%. Sin embargo, es importante notar que esta diferencia fue estadísticamente significativa y que este resultado superó al obtenido con el resultado de referencia (84.7%). Esto indica que nuestro método obtiene una exactitud más alta usando solamente 1,000 instancias etiquetadas de entrenamiento (cien por categoría) en vez de considerar el conjunto de 7,206 ejemplos de entrenamiento.

Con la finalidad de observar el efecto de incorporar nueva información al conjunto de entrenamiento se presentan a continuación las figuras 5.3 y 5.4, las cuales permiten ver, para el caso del conjunto de entrenamiento formado por cien ejemplos de entrenamiento, cómo se van definiendo las categorías en la medida en que se agrega nueva información. En ambas figuras se utilizó un umbral de similitud de 0.4.

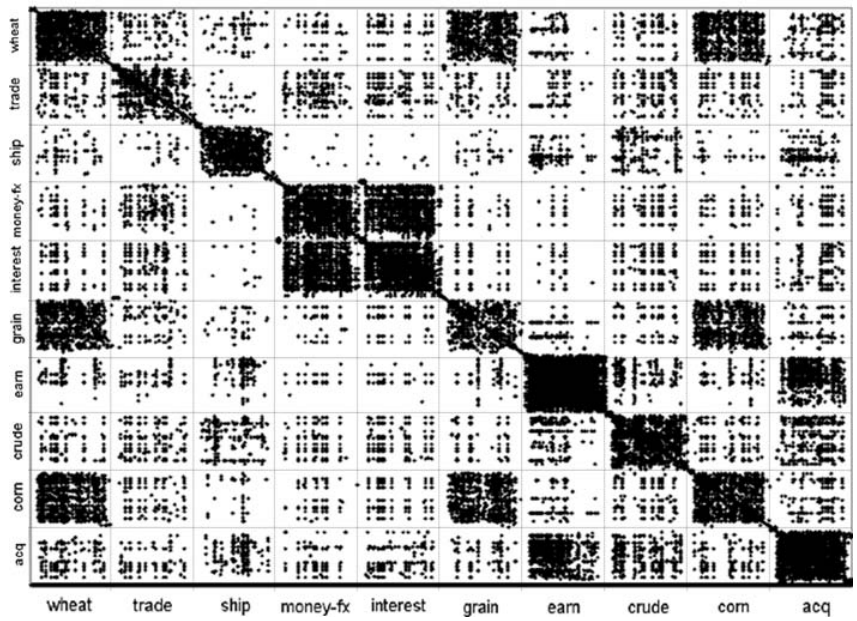

Figura 5.3: Gráfica de similitud corpus de entrenamiento, Reuters

En particular la figura 5.3 muestra el conjunto de entrenamiento formado por cien instancias etiquetadas por categoría. En ella podemos observar la confusión que existe entre las categorías, por ejemplo *money-fx* e *interest*, así como las categorías *corn*, *grain* y *wheat* en las cuales es evidente la confusión del sistema de categorización. Además se puede apreciar en la misma figura la dificultad de categorizar esta colección, ya que la confusión prácticamente existe en todas las categorías, aunque, como se puede observar, en unas un poco más marcada que en otras.

En la figura 5.4 se muestra, para este mismo conjunto de entrenamiento, pero en su sexta iteración, la gráfica de similitud. En ella podemos observar que la confusión entre las categorías ha disminuido significativamente. Sin embargo, aún existe el problema de confusión entre las categorías. Este problema se debe a la cercanía de lenguaje entre los escritos de las diferentes categorías y la consecuente similitud entre ellas.

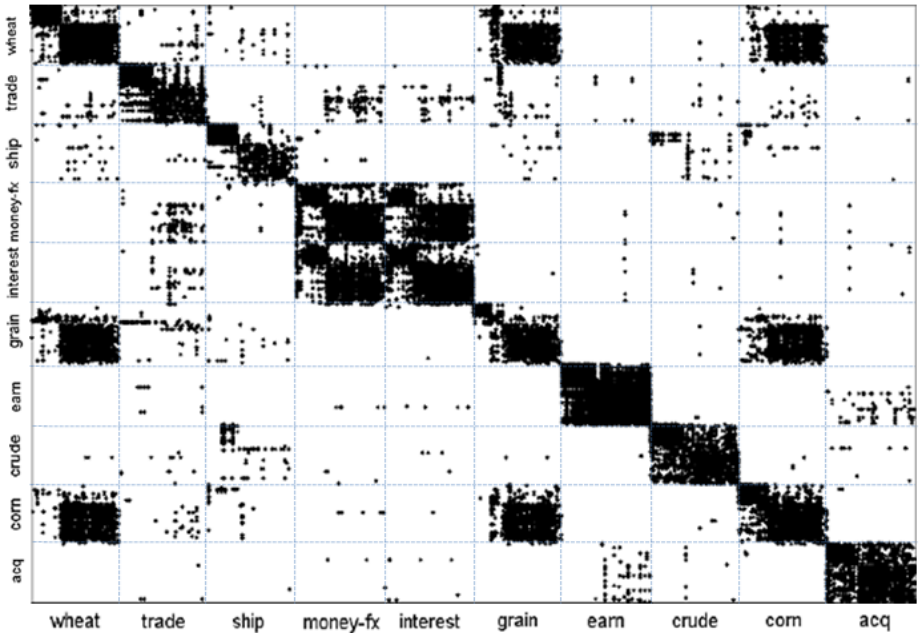

Figura 5.4: Gráfica de similitud corpus de entrenamiento enriquecido, Reuters

En general, se puede decir que los resultados obtenidos son muy relevantes ya que, nos permiten comprobar que con pocos ejemplos de entrenamiento se tienen resultados similares a los obtenidos al usar todo el conjunto de entrenamiento de la colección de Reuters. Además las gráficas de similitud mostradas permiten también ver el impacto que hay en la definición de las categorías al incorporar información no etiquetada proveniente de la Web. Esto se debe a que el método propuesto basado en la Web filtra y etiqueta los ejemplos no etiquetados permitiendo seleccionar las mejores instancias que permitan incrementar la diferencia entre las categorías.

De la misma manera que para el experimento anterior se presenta a continuación el incremento en el vocabulario, así como el valor de la medida *SLMB*. Estos valores se presentan para los corpus mostrados en las gráficas de similitud. Es decir, para el conjunto de entrenamiento formado por cien ejemplos de entrenamiento en su primera y sexta iteración en las cuales se han incorporado respectivamente diez y sesenta ejemplos no etiquetados a cada categoría respectivamente. Como se puede apreciar en la tabla 5.11, el valor de *SLMB* disminuye a la hora de incorporar nueva información al conjunto de entrenamiento. Esto se debe al proceso de selección de la información no etiquetada, ya que la similitud con los documentos originales es grande. Sin embargo, se logra la incorporación de nuevo vocabulario, como se puede apreciar al comparar el número de palabras distintas con el total de palabras en ambas iteraciones.

Tabla 5.11: SLMB y vocabulario para la colección de Reuters.

Iteración	SLMB	PD	TP
1	1994.69	8,933	105,037
6	1047.91	10,083	127,152

5.4 Atribución de autoría de poemas

Los experimentos llevados a cabo en las secciones anteriores corresponden a tareas de categorización temática de textos. En los problemas de categorización temática se cuenta con un conjunto de categorías (previamente definidas) y el trabajo del sistema de categorización consiste en asignar a un documento no visto por el conjunto de entrenamiento a una de estas categorías disponibles en base al vocabulario del mismo.

En este apartado se presentan los resultados obtenidos en un experimento de categorización no temática, en el cual la tarea consistió en identificar al autor de un poema. En este caso, el trabajo consistió en identificar características de escritura que permitieron identificar al autor que escribió el poema en cuestión.

5.4.1 Objetivo del experimento

Otro experimento diseñado para probar el método propuesto en tareas de categorización textual consiste en la atribución de autoría. Esta tarea tiene como finalidad determinar automáticamente el autor que corresponde a un texto anónimo. Es importante comentar que las diferentes aproximaciones existentes para la atribución de autoría (véase capítulo 3 para una descripción de éstas) van desde usar medidas estilométricas y análisis sintáctico hasta métodos que están basados en el uso del vocabulario de los documentos como característica de categorización. Muchas de estas aproximaciones se basan en la idea simple de que para identificar al autor de un texto, el estilo de escritura es más importante que el tópico. En concordancia con esta idea, este experimento de categorización textual fue hecho con la finalidad de determinar si es posible encontrar en la Web información de estilo y si ésta podría ser incorporada al conjunto de entrenamiento con la finalidad de mejorar la exactitud de categorización del método propuesto.

En los siguientes apartados se describen el corpus utilizado en estos experimentos y los resultados obtenidos, tanto con respecto al resultado de referencia como al aplicar el método propuesto.

5.4.2 Descripción del corpus

Debido a que no existen colecciones estándar de datos para evaluar métodos de atribución de autoría, hubo que construir un corpus propio. Este corpus fue generado a partir de la Web y consiste de 353 poemas escritos por cinco diferentes autores. La tabla 5.12 resume algunas cifras acerca de este corpus. Cabe mencionar que este corpus se encuentra disponible[7].

Tabla 5.12: Estadísticas del corpus de poetas

Poeta	Número de documentos	Palabras (tokens) total	Número de frases	Promedio palabras por documento	Promedio frases por documento
Efraín Huerta	48	11,352	510	236.5	22.3
Jaime Sabines	80	12,464	717	155.8	17.4
Octavio Paz	75	12,195	448	162.6	27.2
Rosario Castellanos	80	11,944	727	149.3	16.4
Rubén Bonifaz	70	12,481	720	178.3	17.3

Es importante mencionar que los poemas que forman esta colección son textos muy cortos (172 palabras en promedio) y que todos los poemas corresponden a poetas contemporáneos mexicanos. A manera de ejemplo, tanto del tipo de escritura como de la longitud de los textos, a continuación se presenta un poema de esta colección correspondiente al Octavio Paz.

7 http://ccc.inaoep.mx/~mmontesg/resources/Poetas .sgm

MONÓLOGO

Bajo las rotas columnas,
entre la nada y el sueño,
cruzan mis horas insomnes
las sílabas de tu nombre.

Tu largo pelo rojizo,
relámpago del verano,
vibra con dulce violencia
en la espalda de la noche.

Corriente oscura del sueño
que mana entre ruinas
y te construye de nada:
amargas trenzas, olvido,
húmeda costa nocturna
donde se tiende y golpea
un mar sonámbulo, ciego.

Como se puede apreciar, los poemas manejan una estructura de lenguaje distinta a los textos comunes. Se puede apreciar también el uso de palabras “raras” o que no son de uso muy común. Además la estructura de oraciones se percibe distinta de una línea a otra.

5.4.3 Resultados

Debido a la dificultad de poder comparar el método propuesto con otros métodos utilizados en trabajos previos (ya que no existen corpus estándar de evaluación) fueron diseñados varios experimentos con la finalidad de establecer un resultado de referencia propio. Estos experimentos consideran el uso de 4 diferentes conjuntos de características que se describen a continuación:

Palabras funcionales.- Por palabras funcionales se entiende aquellas palabras que sin tener un significado propio son utilizadas como conectores para integrar un mensaje. Ejemplos de estas palabras son las preposiciones, los artículos y las conjunciones. A pesar de que las palabras funcionales no parecen ser marcas de estilo confiables, ya que son muy frecuentes y ocurren en todo texto, el uso y frecuencia de estas palabras es característico del estilo de los autores. La categorización basada en este tipo de caracterización trabaja apropiadamente, pero es muy sensible al tamaño de los documentos.

Palabras de contenido.- Este es el enfoque tradicional usado para la categorización temática, debido a que normalmente las palabras vacías son eliminadas quedando sólo las palabras de contenido. En algunos casos se acompañan de filtros o medidas del peso de los términos que permitan incrementar la exactitud.

Combinación de palabras funcionales y palabras de contenido.- Este enfoque caracteriza los documentos por un conjunto de secuencias relevantes que combinan palabras funcionales, de contenido y signos de puntuación. La idea es usar estas secuencias para categorizar los documentos, en vista de que éstas expresan las colocaciones léxicas[8] más significativas utilizadas por el autor.

8 Combinaciones frecuentes de unidades léxicas (palabras)

N-gramas de palabras.- Finalmente, un cuarto enfoque considera *n*-gramas, es decir, secuencias de *n* palabras sucesivas. Este enfoque intenta capturar la estructura del lenguaje de los textos por medio de simples secuencias de palabras en contraste con las complejas estructuras sintácticas. De esta manera, el propósito es obtener una adecuada caracterización de los textos, pero sin llevar a cabo un costoso análisis sintáctico. Desafortunadamente este tipo de caracterización nos lleva a una explosión combinatoria, por lo que comúnmente se emplean secuencias de una, dos o a lo más tres palabras (uni-gramas, bi-gramas y tri-gramas).

5.4.3.1 Resultados de referencia

En la tabla 5.13 se muestran los resultados obtenidos al utilizar los cuatro diferentes conjuntos de características basados en palabras descritos en el apartado anterior, donde los promedios corresponden a macro-promedios. Podemos observar que el uso de n-gramas permite un mejor tratamiento de las colocaciones léxicas como “Bill Gates” o “White House”, considerando que la representación anterior (*bolsa de palabras*) separaba estas palabras y su estructura se perdía.

Tabla 5.13: Resultados de referencia atribución de autoría de poemas

Características	Exactitud	Precisión promedio	Recuerdo promedio
Palabras funcionales	41.0	0.42	0.39
Palabras de contenido	73.0	0.78	0.73
Palabras funcionales y palabras de contenido	73.0	0.78	0.74
N-gramas (uni-gramas más bi-gramas)	78.8	0.84	0.79
N-gramas (de uni-gramas a tri-gramas)	76.8	0.84	0.77

Los resultados obtenidos para este experimento son muy interesantes ya que nos permiten ver que:

1. Las palabras funcionales no ayudan a capturar el estilo de escritura de textos cortos.

2. Las palabras de contenido contienen alguna información relevante para distinguir a los autores, cuando todos los documentos corresponden al mismo género y discuten tópicos similares.
3. Las colocaciones léxicas, capturadas por n-gramas de palabras, son útiles para la tarea de identificación de autoría.
4. Dadas las características y el tamaño pequeño del corpus utilizado, el uso de n-gramas grandes (trigramas en particular) no necesariamente ayudan a mejorar la exactitud de categorización.

5.4.3.2 Resultados al aplicar el método propuesto

Para este último experimento se organizó el corpus de diferentes maneras con respecto al experimento del resultado de referencia descrito en el apartado anterior. Específicamente, el corpus fue dividido en dos conjuntos de datos: entrenamiento (con 80% de los ejemplos etiquetados) y prueba (20% de los ejemplos). La idea fue tener un diseño de experimento en las condiciones más reales posibles, cuando no es posible conocer el vocabulario utilizado. Éste es un aspecto muy importante para tomar en cuenta en la atribución de autoría de poemas debido a que los poetas tienden a utilizar un vocabulario muy rico. En la tabla 5.14 se muestran algunos números acerca del conjunto de documentos utilizados como entrenamiento y prueba. El número de palabras distintas que se muestra en la tabla corresponde al conjunto de entrenamiento.

Tabla 5.14: Colección de poetas para la atribución de autoría

Poeta	Conjunto de entrenamiento	Conjunto de prueba	Palabras distintas
Efraín Huerta	38	10	2,827
Jaime Sabines	64	16	2,749
Octavio paz	60	15	2,431
Rosario Castellanos	64	16	3,280
Rubén Bonifaz	56	14	3,552
Total	282	71	8,377

Tomando en cuenta los resultados descritos en el apartado anterior, se decidió utilizar n-gramas como características de los documentos. Fueron diseñados dos experimentos diferentes: en el primero se utilizaron bi-gramas como características de documentos, mientras que en el segundo experimento se utilizaron tri-gramas. La tabla 5.15 muestra los resultados correspondientes de las primeras tres iteraciones del método. Como se puede observar, la integración de nueva información mejora los resultados obtenidos con respecto al resultado de referencia. Sin embargo, en este caso el resultado no es estadísticamente significativo con respecto al resultado de referencia.

Tabla 5.15: Exactitud aplicando el método propuesto a la atribución de autoría.

Características	Exactitud resultado de referencia	Iteración 1ra	Iteración 2da	Iteración 3ra
bi-gramas	78.9	80.3	82.9	80.3
tri-gramas	74.6	74.7	78.8	80.3

Con la finalidad de comprender mejor estos resultados, se presentan a continuación algunas gráficas de similitud correspondientes a esta colección. En todas las gráficas de similitud presentadas en este apartado se utilizó un umbral de 0.2. En particular, en la figura 5.5 se presenta la gráfica de similitud correspondiente al conjunto de entrenamiento formado por palabras funcionales. Como se puede apreciar en la figura, es difícil distinguir las categorías. Esto se debe a la similitud del vocabulario utilizado por los diferentes poetas debido a que, aunque tienen diferentes estilos, todos hablan por ejemplo del amor o de la mujer. Además, podemos observar que existe poca similitud incluso entre los textos correspondientes a un mismo autor. Esto se debe a que seguramente los poetas tienden a escribir acerca de varios tópicos provocando una diferencia en las palabras utilizadas entre dos poemas distintos, disminuyendo la similitud entre ellos. A eso se debe que el sistema confunda incluso los poemas correspondientes a un mismo autor.

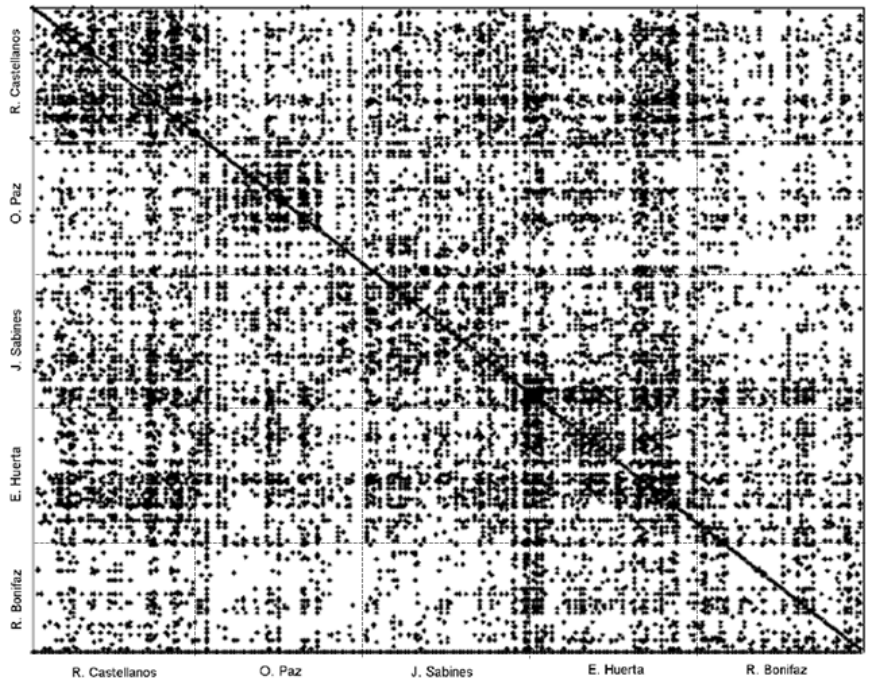

Figura 5.5: Gráfica de similitud usando palabras funcionales, Poetas

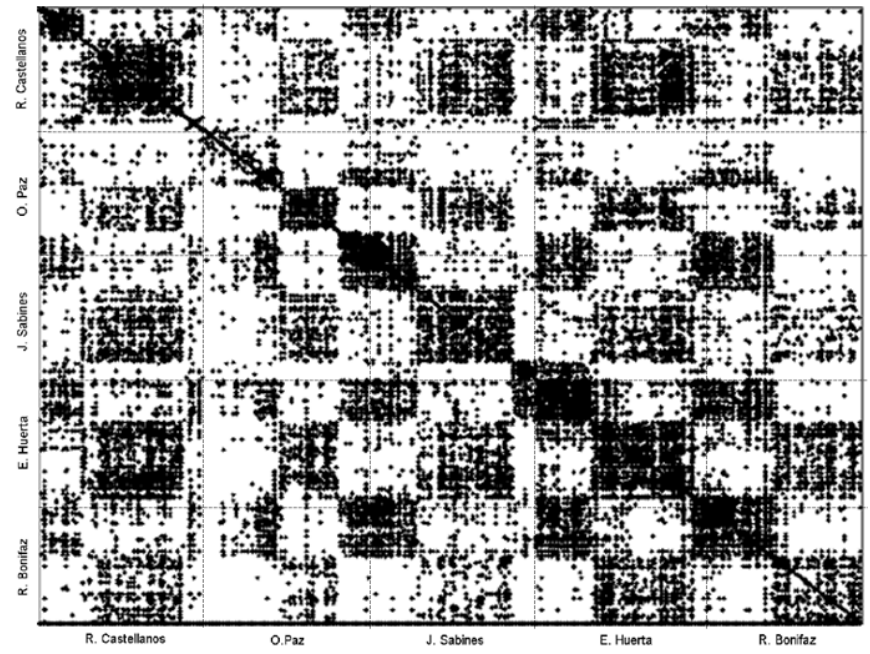

Figura 5.6: Gráfica de similitud usando trigramas, Poetas

En la figura 5.6 se muestra la gráfica de similitud al utilizar el conjunto de entrenamiento formados por tri-gramas en la tercera iteración. Como se puede apreciar, a pesar de que nos permite distinguir mejor las diferentes categorías, la confusión entre las categorías, prácticamente todas, prevalece. Sin embargo, se puede apreciar un cambio significativo respecto a la figura 5.5. Este cambio se debe a la incorporación de información no etiquetada proveniente de la Web que contiene atributos que permiten incrementar la diferencia entre categorías. También podemos observar una mayor similitud de escritura entre los poetas Efraín Huerta y Rosario Castellanos. Este hecho nos permite comprobar que sí se puede encontrar información de estilo en la Web. Cabe mencionar que esta información de estilo es obtenida utilizando el proceso para descargar información de la Web descrito en el capítulo cuatro, esto es, formando las peticiones con las palabras relevantes y utilizando éstas con la máquina de búsqueda para descargar la información.

Por último, en la figura 5.7 se muestra la gráfica de similitud al utilizar el conjunto de entrenamiento formado por bi-gramas en la tercer iteración. En esta figura es interesante ver como los grupos de datos que representan a los diferentes poetas se vuelven más compactos, aunque la confusión persiste. Sin embargo, nos permite observar un mejor desempeño al poder identificar sobre la diagonal principal de la matriz los grupos definidos. Esto habla del estilo de escritura de los poetas, al poder identificar secuencias de palabras de tamaño dos que se repiten con mayor frecuencia y es lo que hace que sea más “distintivo” entre un poeta y otro. También se puede identificar prácticamente dos grupos para cada poeta. Esto se puede relacionar con el uso repetido de secuencias de tamaño dos independientemente del tema o tópico tratado en el poema.

Figura 5.7: Gráfica de similitud usando bigramas, poetas

Dados estos resultados preliminares, podemos concluir que es posible extraer información de estilo de la Web para llevar a cabo la tarea de atribución de autoría. No obstante, la intuición sugiere lo contrario: dado que los poemas tienden a tener una combinación inusual de palabras y la Web no parece, en primera instancia, ser una fuente adecuada de información relevante para esta tarea. Al contrario, estos resultados preliminares nos muestran que es posible extraer información relevante de la Web para llevar a cabo también una tarea tan específica como la atribución de autoría.

Con la finalidad de comprender aun mejor los resultados obtenidos, se presenta a continuación, en la tabla 5.16 los resultados obtenidos para estos conjuntos de entrenamiento de la medida *SLMB* y del vocabulario, así como del grado de desbalanceo entre las categorías.

Tabla 5.16: Medidas SLMB y vocabulario para el corpus de poetas.

Conjunto	SLMB	Desbalanceo	PD	TP
Palabras funcionales	26.25	0.03	8,378	47,363
Tri-gramas	29.13	0.02	9,320	52,707
Bi-gramas	35.15	0.02	9,916	56,023

Como se puede apreciar en la tabla 5.16, se logra la incorporación de nuevo vocabulario en las iteraciones al llevar a cabo la categorización de la colección de poetas, aunque en una escala menor con respecto a los experimentos de categorización temática descritos en las secciones anteriores.

5.5. Conclusiones del capítulo

En este capítulo se presentaron los resultados obtenidos en la tarea de categorización automática de documentos utilizando el método semi-supervisado basado en la Web. Se llevaron a cabo tres experimentos de categorización de textos: noticias sobre desastres naturales en español, noticias de la colección Reuters en inglés, y la atribución de autoría de poemas. Los tres experimentos fueron llevados a cabo utilizando una configuración común, la cual consistió en descargar el mismo número de snippets para cada categoría (mil snippets por categoría), aplicar el mismo pre-procesamiento, así como los algoritmos de categorización y medidas de evaluación utilizados. Los experimentos realizados permiten ver la funcionalidad del método propuesto, al poder utilizarlo en contextos de aplicación muy diferentes entre sí, ya que, por ejemplo, en el caso de las noticias sobre desastres naturales se formaron conjuntos de entrenamiento con menos de diez ejemplos para cada categoría. Éste es un hecho relevante, considerando que el número de instancias de entrenamiento es muy pequeño para que un sistema de aprendizaje automático logre aprender, considerando que los métodos tradicionales de categorización se caracterizan precisamente por requerir un gran número de instancias de entrenamiento, comúnmente cientos o miles.

El experimento basado en las noticias de la colección de Reuters se caracteriza por tener un alto grado de desbalanceo y traslape entre las categorías. Sin embargo, y a pesar de la dificultad de la tarea, se logró superar los resultados de referencia obtenidos al incorporar información no etiquetada de la Web al conjunto de entrenamiento.

Por último, la atribución de autoría presenta el caso de una categorización basada en estilo. Los resultados obtenidos permiten ver la efectividad del método en estos escenarios tan distintos. En el estado del arte cada uno de estos problemas representa una línea de investigación abierta, caracterizándose por desarrollar sistemas de categorización dedicados únicamente a la solución de una de estas problemáticas a la vez. Como se puede observar en los resultados obtenidos, en todos los casos se formó un conjunto de prueba independiente, esto es, nunca fue visto por el conjunto de entrenamiento y se logró mejorar los resultados obtenidos como referencia.

Capítulo VI

Desambiguación del sentido de las palabras

Los experimentos descritos en el capítulo anterior, referentes a la categorización automática de documentos tanto temática como no temática, mostraron que el método desarrollado permite obtener buenos resultados cuando se utilizan textos relativamente cortos (por ejemplo, las noticias de desastres naturales en el caso de la categorización temática y los poemas en el caso de la categorización no temática). Este hecho también ocurre en la tarea de la desambiguación del sentido de las palabras, la cual se caracteriza por tener contextos de ocurrencia, de una palabra a ser desambiguada, generalmente cortos.

Así, en este capítulo se presentan los resultados obtenidos al aplicar el método semi-supervisado basado en self-training y la Web a la tarea de la desambiguación del sentido de las palabras. El método fue probado utilizando un corpus formado por un subconjunto de sustantivos pertenecientes a la tarea *English lexical sample* de *SemEval*[1]. En los apartados siguientes se presenta una descripción de la tarea, así como los principales enfoques utilizados para dar solución a esta problemática. Se presenta, también, la evaluación requerida en esta tarea, así como los resultados experimentales obtenidos utilizando el corpus mencionado tanto de referencia como al aplicar el método propuesto.

1 http://nlp.cs.swarthmore.edu/semeval/tasks/task17/description.shtml

6.1 Descripción de la tarea

La desambiguación del sentido de las palabras (Word Sense Disambiguation, WSD) es el problema de decidir cuál es el sentido correcto de una palabra en un contexto determinado y es una de las tareas primordiales en muchas de las aplicaciones del procesamiento del lenguaje natural (Navagli, 2009; Agirre et al., 1996). Por ejemplo, dado el enunciado "*El tomó su dinero del banco*", si nos enfocamos a la palabra *banco* el objetivo sería identificar el sentido deseado, el cual en este texto sería de carácter "financiero", en lugar de alguna otra posibilidad como asiento o algún establecimiento médico donde se conservan y almacenan órganos, o algún otro.

Los sentidos pueden ser definidos en un diccionario, una base de conocimientos léxicos o una ontología. Esta tarea es definida como un paso intermedio hacia el entendimiento del lenguaje natural (Duffield et al., 2007). La construcción de algoritmos eficientes para llevar a cabo la tarea de WSD podrían beneficiar varias aplicaciones del procesamiento del lenguaje natural, tales como traducción de automática (Aguirre et al., 2007) y recuperación de información (Resnik et al., 2005). Siguiendo con la frase de ejemplo, si se utiliza un sistema de traducción automática para poner esta frase en francés, el sistema de desambiguación deberá ser capaz de proporcionar la palabra "banque" cuando es usada en el sentido financiero descartando otras posibles traducciones de la palaba. En el caso de sistemas de recuperación de información podría ser útil determinar cuál es el sentido de una palabra a fin de recuperar documentos relevantes a una petición en particular. Por ejemplo, si se pretende recuperar imágenes en base a sus pies de imagen, esta tarea adquiere una mayor relevancia cuando se trabaja en más de un idioma dado que se tendría acceso potencialmente a más resultados relevantes (Zeman et al., 2008).

Así como se ha comentado anteriormente, la asociación de una palabra, a un sentido, depende de dos tipos de recursos de información: el contexto y los recursos léxicos de conocimientos externos. El contexto del sustantivo a ser desambiguado se define como el conjunto de palabras de la misma frase. Existen bases de datos léxicas, como WordNet[2] para el idioma inglés, que se pueden usar como recurso léxico de conocimiento externo. El conocimiento lingüístico puede ayudar al proceso de WSD no sólo como aportación teórica, de orden general, sobre el lenguaje y las lenguas, sino también como información particular, relacionada con el uso de las palabras en un contexto particular. Así, la investigación en WSD debe tener una visión más consistente con la teoría lingüística (Navigli, 2009; Agirre et al., 1996). Además, se debe explotar la visión complementaria de la lingüística del corpus, fundada en grandes cantidades de texto, que aportan datos concretos sobre las características individuales de las palabras a desambiguar.

Los métodos automáticos desarrollados para llevar a cabo la desambiguación del sentido de las palabras son muy diversos. Existen estudios que presentan una clasificación detallada (Navigli, 2009) o que exponen los problemas que se presentan en la evaluación de dichos métodos (Resnik et al., 2000; Aguirre et al., 2007). A pesar del tiempo y esfuerzo dedicados en esta dirección lo cierto es que hasta la fecha no se ha conseguido desarrollar ningún sistema de amplia cobertura con resultados plenamente satisfactorios. Pero, ¿Qué se espera de un sistema automático de desambiguación del sentido de las palabras? A manera de responder esta pregunta a continuación se presenta un ejemplo. En la figura 6.1 se muestra la definición de los dos diferentes sentidos de la palabra *age*. Se presenta, además, un ejemplo para cada sentido. La idea es que se pueda visualizar claramente la complejidad de la tarea. Un sistema de WSD deberá ser capaz de decidir el sentido correcto de la palabra *age* en cada ejemplo, usando la información del contexto en el que aparece. Como se puede intuir, ésta no es una tarea trivial aún con la ayuda de una computadora.

2 http://wordnet.princeton.edu/

Definiciones de los sentidos	
Age-1	The length of time something (or someone) has existed; "his age was 71"; "it was replaced because of its age"
Age-2	A historic period; "the Victorian age"; "we live in a litigious age"
Frases de ejemplo	
Age-1	He was mad about stars at the <age> of nine.
Age-2	About 20,000 years ago the last ice <age> ended.

Figura 6.1: Ambigüedad de la palabra *age*

La dificultad para abordar la tarea de WSD se debe a varias razones. Por un lado están las dificultades intrínsecas de la tarea, entre las que destacan:

- El grado de ambigüedad medio por palabra es mayor que en otras tareas, debido a que el grado de ambigüedad en el etiquetado morfosintáctico está entre 2 y 3 etiquetas por palabra, mientras que en WSD puede ser entre 5 y 6 sentidos por palabra.

- El contexto necesario para poder desambiguar una palabra puede ser muy extenso, llegando incluso a requerir párrafos u oraciones anteriores.

- Las fuentes de información necesarias para poder desambiguar una palabra son muy diversas, entre las que destacan: morfología, sintaxis y conocimiento pragmático. Al ser tantas y tan variadas, en ocasiones no se dispone de ellas.

Por otro lado, se encuentran las dificultades en la evaluación de los sistemas de desambiguación: la definición de sentidos utilizada (o diccionario), la lengua, las medidas de evaluación y el alcance del sistema. Los recursos disponibles hasta el momento son insuficientes para que las aproximaciones de aprendizaje automático alcancen unos resultados satisfactorios. Además, la mayoría de estos recursos se han construido principalmente para el inglés, por lo que muchas veces en el resto de las lenguas se ven obligados a desarrollar principalmente aproximaciones no supervisadas.

En el siguiente apartado se presenta una descripción de las técnicas de evaluación desarrolladas así como de competenciones y foros dedicados a este fin. También se presenta una clasificación de los diferentes métodos utilizados en WSD. Una clasificación común de estas aproximaciones es en dos categorías muy generales: métodos basados en conocimiento (*knowledge-based methods*) y métodos basados en corpus (*corpus-based methods*). Los primeros hacen uso del conocimiento adquirido en forma de diccionarios, tesauros, lexicones y ontologías entre otros recursos. Podemos decir que este conocimiento es preexistente al proceso de desambiguación y, en la mayoría de los casos, adquirido de forma manual. Los segundos extraen el conocimiento de grandes cantidades de ejemplos (corpus) mediante métodos estadísticos y aprendizaje automático. En la siguiente sección se presenta una descripción más detallada de los principales métodos utilizados para llevar a cabo la tarea de WSD.

6.2 Evaluación y métodos utilizados en WSD

En esta sección se describe, primeramente la evaluación de un sistema de WSD. Se presenta un apartado de evaluación en general y un segundo apartado que trata acerca de la evaluación directa de los métodos de WSD. Después de presentar una clasificación simplificada del gran número de métodos y soluciones propuestos actualmente para esta tarea. Para cada uno de ellos se presentan trabajos relevantes sin pretender ser una exploración exhaustiva del estado del arte. A continuación se presenta una descripción de estas aproximaciones.

6.2.1 Evaluación

En 1997, Resnik y Yarowsky realizaron una serie de propuestas orientadas a la estandarización de la evaluación en desambiguación (Resnik et al., 1997). Estas propuestas se han plasmado en la celebración de una competición sobre desambiguación llamada en sus orígenes *SENSEVAL* (kilgarriff, 1998) y desde 2007 llamado *SemEval*[3]. El principal objetivo de estas competiciones es la organización de tareas de evaluación y validación de sistemas de WSD respecto a la desambiguación del sentido de determinadas palabras, diferentes aspectos de un idioma, distintos idiomas y diferentes aplicaciones. *SemEval* está organizado por un comité auspiciado por el *ACL-SIGLEX* (el grupo de interés especial en lexicón de la ACL). Hasta la fecha se han realizado cuatro ejercicios de evaluación en los años 1998 (*SENSEVAL-1*), 2001 (*SENSEVAL -2*), 2004 (*SENSEVAL-3*) y 2007 (*SemEval*).

En la última edición se llevaron a cabo 18 tareas de las 19 programadas[4]. Entre estas tareas se encuentran: *all-words*, *lexical-sample* y *translation*. Estas tareas se describen brevemente a continuación:

3 http://nlp.cs.swarthmore.edu/semeval/
4 http://nlp.cs.swarthmore.edu/semeval/tasks/index.php

- *all-words*: los sistemas deben etiquetar todas las palabras con contenido semántico de un texto.
- *lexical-sample*: tiene como objetivo evaluar solamente un conjunto previamente seleccionado de palabras.
- *translation*: un sentido de una palabra se define de acuerdo a sus distintas traducciones.

Como se puede apreciar, estas competiciones están enfocadas a la evaluación directa de la desambiguación. La evaluación directa mide la efectividad en la asignación de los significados correctos a las palabras a desambiguar. La evaluación directa es fundamental para cuantificar la calidad de los distintos enfoques de desambiguación. Sin embargo, las evaluaciones directas realizadas suelen presentar los siguientes problemas:

- La falta de acuerdo en la elección de las definiciones de las palabras: diferentes diccionarios suelen dar distintos conjuntos de sentidos para la misma palabra.
- La escasez de colecciones de evaluación: el etiquetado semántico de un corpus es una tarea difícil y costosa.
- La inconsistencia en el etiquetado de las colecciones de evaluación: distintas personas pueden asignar diferentes significados a la misma palabra en el mismo contexto.
- Una cierta falta de acuerdo en las métricas utilizadas: diversos autores presentan diferentes formas de medir la efectividad de la desambiguación.

Cabe mencionar que el hecho de llevar a cabo estas competiciones SENSEVAL y SemEval ha permitido la solución parcial de estas problemáticas ya que han propiciado el desarrollo de conjuntos de evaluación únicos para la competición así como el conjunto de significados adaptados a la colección.

También se lleva a cabo la evaluación indirecta de la desambiguación, la cual se asocia a la evaluación propia de la tarea en la cual se aplica la desambiguación como puede ser categorización de textos o recuperación de información, donde en función de la efectividad de estas tareas se lleva a cabo la evaluación a la desambiguación.

La evaluación de sistemas WSD ha supuesto un problema en la investigación de la desambiguación del sentido de las palabras. De hecho, algunos sistemas de desambiaguación han sido evaluados sólo a través de pruebas manuales sobre un grupo reducido de palabras. La evaluación es muy heterogénea. En numerosos trabajos se han utilizado diferentes métricas y colecciones de prueba. Algunas de estas medidas de evaluación son heredadas de la evaluación de los sistemas de recuperación de información (Salton et al., 1983; Frakes et al., 1992). En nuestro caso la evaluación se lleva a cabo utilizando una colección de prueba, la cual no ha sido vista nunca por el conjunto de entrenamiento. Esta manera de llevar a cabo la evaluación es el escenario más deseable, ya que permite identificar o asociar el sentido a una palabra polisémica en función del contexto pero en este caso el conjunto de prueba no ha sido visto por el conjunto de entrenamiento. Como métrica básica para evaluar la efectividad de nuestro sistema de desambiguación se tomó la exactitud (Lewis et al., 1994). La exactitud puede ser definida como el cociente entre el número de términos desambiguados satisfactoriamente y el número de términos desambiguados. En los siguientes apartados se presenta una descripción de los principales métodos de desambiguación existentes.

6.2.2 Métodos basados en conocimiento

La escasez de corpus etiquetados semánticamente es un gran problema en la tarea de WSD. Las aproximaciones basadas en conocimiento tienen la ventaja de no requerir procesos de entrenamiento, además no necesitan etiquetado manual. Estos métodos generalmente utilizan la información que se encuentra almacenada en algún recurso (diccionarios, tesauros o bases de datos léxicas).

El uso de diccionarios electrónicos se inició con los trabajos de Lesk, quien creó una base de conocimiento que asoció con cada sentido en un diccionario. La desambiguación se realizaba seleccionando el sentido de la palabra que tenía mayor número de traslapes en las palabras vecinas del contexto. El método logró entre un 50 y un 70 % de palabras desambiguadas correctamente (Lesk, 1986). Este método es muy sensible a la redacción exacta de cada definición y ha sido usado como base para los trabajos posteriores que se han realizado en el área.

El conocimiento es un componente fundamental de WSD. Las fuentes de conocimiento proporcionan datos que son esenciales para asociar sentidos con las palabras. Existen distintas aproximaciones estadísticas que calculan la probabilidad de asignar a una palabra un determinado sentido, según el contexto en el que aparezca. Algunos de estos sistemas requieren fuentes de conocimiento externo (Marquez et al., 2006).

Las fuentes de conocimiento pueden ser clasificadas en dos grandes grupos:

- Estructuradas y
- No estructuradas.

A continuación se describe cada una de ellas, así como las fuentes de conocimiento más representativas de cada una de ellas.

Recursos estructurados

- **Tesauros**: Proporcionan información acerca de las relaciones entre las palabras, tales como sinonimia (por ejemplo, auto es sinónimo de vehículo), antonimia (representa el sentido opuesto, por ejemplo bonito es antónimo de feo) así como posibles futuras relaciones (Killgarriff et al., 2000). El tesauro más común en el área de WSD es sin duda el tesauro Roget´s, *Roget´s International Thesaurus* (Roget 1977). La última edición de este tesauro contiene 250,000 palabras organizadas en seis tópicos generales y cerca de 1,000 categorías.

- **Diccionarios**: Comúnmente conocidos como MRD´s, *Machine Readable Dictionaries*, son una fuente popular de conocimiento para tareas relacionadas con el procesamiento del lenguaje natural desde que el primer diccionario estuvo disponible en formato electrónico en los años 80´s. Después de éste, se encuentran disponibles una gran variedad de diccionarios, entre los que destacan: Collins English Dictionary, Oxford Advanced Learner´s Dictionary of Current English y el Oxford Dictionary of English (Soanes et al., 2003), además está el Dictionary of Contemporary English (LDOCE) (Proctor, 1978). Existen trabajos que han utilizado el diccionario LDOCE en conjunto con WordNet para llevar a cabo tareas de WSD. Entre estos trabajos destacan los siguientes (Miller et al., 1990; Fellbaum,1998)

- **Ontologías**: contienen especificaciones de conceptos de un dominio específico de interés (Gruber, 1993) y usualmente incluyen una taxonomía y un conjunto de relaciones semánticas. En este sentido, WordNet puede ser considerado como una ontología. También destaca la *Omega Ontology* (Philpot et al., 2005), caracterizada por reorganizar y conceptualizar WordNet, y la ontología *SUMO upper Ontology* (Pease et al., 2002).

Recursos no estructurados

- **Corpus**: Esto es, una colección de texto utilizada para aprender modelos de lenguaje. Los corpus pueden ser anotados por sentido de las palabras o no. Sin embargo, ambos son muy utilizados tanto en aproximaciones supervisadas como no supervisadas en tareas de WSD. Los corpus se dividen en tres g*randes grupos:* raw corpora, corpus con sentidos anotados y recursos con colocaciones, cada uno de ellos con aplicaciones muy interesantes en WSD. Algunos de los corpus más utilizados se describen a continuación:
    - **Raw corpus**: Dentro de esta clasificación de corpus se encuentran los siguientes: el corpus Brown (Kucera et al., 1967) que es una colección balanceada de un millón de palabras. Corresponden a textos que fueron publicados en los Estados Unidos en 1961; el BNC, British National Corpus (Clear, 1993), el cual es una colección de cien millones de palabras de ejemplos escritos y hablados en idioma inglés (comúnmente utilizado para recopilar secuencias de palabras e identificar relaciones gramaticales entre palabras); el corpus del periódico Wall Street Journal, (Charniak et al., 2000), el cual es una colección de aproximadamente 30 millones de palabras. El *American National Corpus* (Ide, 2006), el cual incluye 22 millones de palabras en inglés, y por último, el corpus más grande hecho hasta ahora, el *Gigaword Corpus*, una colección de 2 billones de palabras de noticias (Graff, 2003).
    - **Corpus con sentidos anotados**: SemCor (*SEMantic COncoRdance*) fue construido sobre un fragmento del corpus *Brown* y de la novela *The Red Badge of Courage* de Stephen Craig, dentro del proyecto WordNet (Miller et al., 1993). Es el más grande y usado corpus con sentidos etiquetados e incluye más de 352 textos etiquetados con alrededor de 350,000 palabras (cada palabra está etiquetada con un concepto de WordNet).

Este etiquetado posibilita la evaluación de los algoritmos de WSD para todas las palabras. Sin embargo, aunque cubre un gran número de palabras, contiene un conjunto muy bajo de ejemplos para cada una de ellas, lo cual es una limitante muy importante para su uso. SemCor es el único corpus disponible libremente con todas las palabras de clase abierta etiquetadas. MultiSemCor (Pianta et al., 2002) es un corpus paralelo anotado para el inglés e italiano, basada en la versión italiana de Wordnet.

- **Recursos de colocaciones**: Contienen registros de tendencias de las ocurrencias de palabras de uso común o regulares con otras palabras. Como ejemplo de este tipo de recursos, se tienen los siguientes: Word Sketch Engine[5], Just The Word[6], The British National Corpus Collocations[7] y el Collins Cobuild Corpus Concordance[8]. Recientemente se ha desarrollado un sistema que ha ganado popularidad entre la comunidad que desarrolla sistemas para la tarea de WSD: el sistema Web1TCorpus (Brants et al., 2006). Este corpus contiene frecuencias para secuencias de más de cinco palabras en un corpus de un trillón de palabras obtenidas de la Web.

A continuación se comentan algunos de los trabajos más destacados que utilizan alguna fuente de conocimiento de las descritas en los párrafos anteriores. El funcionamiento básico de estos métodos consiste en medir la similitud entre el contexto en que aparece una palabra y sus definiciones en la fuente de conocimiento. Una aproximación muy interesante es la de Montoyo (Montoyo, 2002). En este trabajo, se alimenta al sistema de desambiguación con un conjunto de palabras. Básicamente es un sustantivo y su contexto. El sistema busca cada una de las palabras que forman el contexto del sustantivo en una base de conocimientos léxica, en este caso *WordNet*, a las cuales aplica el método de marcas de especificidad y de esta manera produce la salida con los posibles sentidos. Este sistema no depende del dominio ni de la lengua.

5http://www.sketchengine.co.uk/
6http://193.133.140.102/JustTheWord/
7http://www.natcorp.ox.ac.uk/
8http://www.collins.co.uk/Corpus/CorpusSearch.aspx

Los sistemas basados en tesauros (un tesauro clasifica las palabras dentro de categorías) parten de la idea de que una palabra que está clasificada en distintas categorías presenta sentidos diferentes en cada una de las categorías. Estos sistemas necesitan conocer el contexto en el que aparece una palabra para poder clasificarla correctamente. Por ejemplo, Yarowsky (Yarowsky, 1992) utilizó el tesauro Roget[9] (en inglés) y hace la extracción del contexto a partir de las definiciones de las palabras a desambiguar presentes en una enciclopedia. Los trabajos de Rada Mihalcea (Mihalcea, 2004) y Aguirre (Aguirre et al., 1996) propusieron fórmulas de distancia conceptual en las que se tiene en cuenta básicamente la longitud del camino entre dos conceptos según las relaciones de hipónimia en WordNet. Mientras que Sussna (Sussna, 1993) introdujo una medida ponderada según el tipo de relación (sinonimia o hipónimia). En la aproximación propuesta por Rosso (Rosso et al., 2003) se propuso una fórmula para el cálculo de la densidad conceptual, así como en (Aguirre et al., 1996) pero considerando sólo los synsets relevantes, es decir, aquellos nodos terminales de los caminos del nombre a desambiguar y de los sustantivos de su contexto, y por los cuales hay que calcular la densidad. Información adicional acerca de los métodos no supervisados utilizados en WSD puede ser encontrada en (Schütze, 1998; Pedersen, 2006; Navigli, 2009).

6.2.3 Métodos basados en corpus: supervisados, no supervisados y semi-supervisados

Dentro de las aproximaciones basadas en corpus tenemos, el modelo de máxima entropía (Suárez et al., 2002). Otra aproximación aplicada a WSD, es basada en los modelos ocultos de Markov (Molina, 2004). Información adicional acerca de los métodos supervisados utilizados en WSD puede ser encontrada en (Schütze, 1998; Jurafsky et al., 2000; Marquez et al., 2006; Navigli, 2009).

9 http://poets.notredame.ac.jp/Roget/

Existe un acuerdo más o menos amplio en que la falta de un corpus apropiado y suficientemente grande representa un obstáculo para continuar progresando en el área de WSD. Es difícil conseguir un corpus anotado con sentidos para aprendizaje automático (Navigli, 2009; Edmonds, 2000; Mihalcea, 2004), y los avances y esfuerzos recientes en su adquisición automática no hacen más que reforzar su importancia para este desarrollo crucial. Debido a este hecho, los métodos semi-supervisados o mínimamente supervisados están ganado popularidad ya que sólo requieren un número pequeño de datos etiquetados logrando mejores valores de exactitud que los métodos basados en sistemas supervisados, los cuales requieren un gran conjunto de datos etiquetados. En los últimos años, han surgido varios trabajos orientados a reducir el costo de adquisición, la necesidad de supervisión y los requerimientos computacionales de los métodos basados en corpus. Así, en estos momentos se encuentran abiertas las siguientes líneas de investigación en el área de WSD:

- El diseño de métodos para construir de manera eficiente muestras de aprendizaje representativas (sampling).
- El uso de recursos léxicos externos, tales como WordNet, y los motores de búsqueda en Internet con el objetivo de extraer ejemplos de aprendizaje automáticamente sin necesidad de anotación manual.
- El uso de algoritmos semi-supervisados que permitan reestimar iterativamente los parámetros estadísticos del modelo sin necesidad de disponer de grandes corpus totalmente etiquetados.

Los metodos semi-supervisados para WSD se han caracterizado por utilizar datos no etiquetados en el proceso de aprendizaje. Los algoritmos iterativo-incrementales (*bootstrapping*) parecen una buena opción para esa adquisición automática de nuevos conjuntos etiquetados de entrenamiento (Mihalcea, 2004; Zhu, 2005; Chapelle et al., 2006). Básicamente, sólo se necesitan unos pocos ejemplos para iniciar un proceso iterativo que se retroalimente, a partir de un conjunto no etiquetado, en sucesivos ciclos de aprendizaje y categorización. Esto es, en cada iteración de bootstrapping los ejemplos no etiquetados son categorizados usando un modelo de aprendizaje formado a partir de los datos etiquetados.

Uno de los primeros métodos incrementales que se citan específicamente para WSD es el de Yarowsky (Yarowsky, 1995). Se trata de un método no supervisado que se basa en dos restricciones: que una palabra tiende a tener un único sentido dentro de un mismo discurso, y también dentro de una misma "colocación" (*one sense per discourse*, *one sense per collocation*). El método se evaluó sobre un pequeño conjunto de palabras con dos posibles sentidos cada una. Partiendo de las definiciones de un diccionario, se construyó una semilla con colocaciones representativas de cada sentido y se utilizó como entrada para un algoritmo de listas de decisión. El incremento del corpus anotado se hacía con aquellas instancias categorizadas que superaban un cierto umbral de probabilidad. Este trabajo tuvo, y tiene un gran impacto en la comunidad científica especializada en WSD por la alta precisión conseguida (95% aproximadamente), aunque es evidente que las condiciones del experimento están un poco alejadas de la realidad.

Rada Mihalcea publicó un estudio comparativo entre co-training y self-training (Mihalcea, 2004) y su aplicación a la tarea de WSD. Cabe mencionar que en este trabajo descargan información de la Web, sin embargo, para poder llevar a cabo esta tarea requiere de información sobre los sinónimos de las palabras a desambiguar. La información de los sinónimos de la palabra ambigua es utilizada para formar peticiones que serán lanzadas a la Web. La información descargada es incorporada al conjunto de entrenamiento sin llevar a cabo ningún proceso de selección de las mejores instancias por medio, por ejemplo, de algún proceso iterativo.

Recientemente se han desarrollado una serie de algoritmos basados en aprendizaje semi-supervisado para llevar a cabo la tarea de WSD (Navigli, 2009) los cuales pueden combinar de una manera efectiva los datos no etiquetados con los datos etiquetados en el proceso de aprendizaje incrementando el conjunto de entrenamiento (Guzmán et al., 2009; Guzmán et al., 2009b). Tales métodos realizan la categorización por medio de la coherencia global y basados en la siguiente hipótesis: ejemplos similares deben tener etiquetas similares. En otras palabras, las etiquetas de los ejemplos no etiquetados son determinadas teniendo en cuenta no sólo la similitud entre la etiqueta y los ejemplos no etiquetados, sino también la similitud entre las etiquetas de ejemplo.

6.3 Resultados experimentales

En esta sección se presentan los resultados experimentales obtenidos con el método semi-supervisado basado en la Web propuesto en esta tesis. A diferencia de los métodos descritos en el apartado anterior, nuestro método tiene la limitante de requerir instancias de entrenamiento en todos los sentidos de la palabra polisémica que se desea desambiguar para formar las peticiones y descargar información no etiquetada de la Web. Sin embargo, a diferencia de otros métodos, éste no requiere de un conjunto previo de información no etiquetada, ya que el método propuesto va a la Web y descarga los ejemplos no etiquetados. La información descargada pasa por un proceso de aprendizaje semi-supervisado que permite la selección de las mejores instancias, las cuales serán incorporadas al conjunto de entrenamiento. En los apartados siguientes se muestran los resultados obtenidos, tanto de referencia como al aplicar el método propuesto.

6.3.1 Objetivo del experimento

El presente experimento está enfocado a la categorización de los sentidos de palabras polisémicas. El objetivo fue evaluar el método propuesto en una tarea que se caracteriza por la dificultad que presenta, debido a que generalmente cuenta con clases desbalancedas y muy pocos ejemplos de entrenamiento. Aunado a esto, los ejemplos de entrenamiento en esta tarea se caracterizan por ser textos muy cortos.

El llevar a cabo este experimento nos permitirá evaluar si es posible incrementar la precisión de categorización en la tarea de WSD al agregar información no etiquetada proveniente de la Web al conjunto de entrenamiento. A continuación se presenta una descripción de la configuración utilizada para el presente experimento, el corpus utilizado y los resultados obtenidos, tanto de referencia como al aplicar el método propuesto.

6.3.2 Configuración del experimento

El experimento llevado a cabo consiste en la aplicación del método descrito en el capítulo 4 a la tarea de WSD. La parte común de los experimentos descritos en la sección 5.1, se utiliza también para este experimento. Fueron descargados 1,000 snippets para cada sentido de la palabra polisémica que se desea desambiguar. Los snippets descargados pasan por la etapa de pre-procesamiento, en la que son eliminados las etiquetas y caracteres especiales. Para seleccionar los snippets que se incorporarán al conjunto de entrenamiento, se utilizó un meta-clasificador. Específicamente se utilizó un arreglo tipo stacking basado en dos clasificadores: naïve Bayes y SVM. Este arreglo es entrenado con el mismo conjunto de entrenamiento utilizado para categorizar al conjunto de prueba. Los snippets con la más alta probabilidad asignada por el arreglo stacking son incorporados al conjunto de entrenamiento. La efectividad del método fue medida a través de la exactitud de categorización, la cual indica el porcentaje de sentidos de una palabra polisémica a las que les fue asignado el sentido correcto de la colección de prueba.

6.3.3. Descripción del corpus

La evaluación del método fue llevada a cabo con un subconjunto de sustantivos correspondientes a la tarea *lexical sample* de la competición SemEval. En particular, se consideraron nueve sustantivos. En la tabla 6.1 se muestran algunas estadísticas del corpus utilizado. Como se puede apreciar en estos sustantivos existe un importante problema de desbalanceo entre los sentidos. Para tener un valor cuantificable de este hecho se muestra el valor de la desviación estándar entre el número de instancias de entrenamiento por sentido. Entre mayor es el valor de la desviación estándar, mayor es el problema de desbalanceo.

Tabla 6.1: Estadísticas del corpus de entrenamiento, SemEval

Sustantivo	Número de sentidos	Ejemplos de entrenamiento	Ejemplos de prueba	Desviación estándar
source	5	151	35	20.64
bill	2	739	114	446.18
president	3	872	176	401.24
management	2	277	44	40.30
condition	2	130	33	59.40
policy	2	329	39	129.4
rate	2	1003	145	490.02
drug	2	205	46	28.99
state	3	609	70	263.03

Por ejemplo, el primer sentido de *bill* tiene 685 instancias de entrenamiento, mientras que el segundo sólo tiene 54. Estos valores tienen una desviación estándar de 446.18, lo cual nos indica un alto grado de desbalanceo. Sin embargo, dado que las peticiones se forman por sentido independientemente de este desbalanceo, el método propuesto puede ser aplicado para enriquecer el conjunto de entrenamiento, considerando que una limitante del método propuesto basado en la Web es que se tengan ejemplos de entrenamiento en todos los sentidos. Los ejemplos de entrenamiento son necesarios para poder obtener las palabras relevantes y formar entonces las peticiones para a su vez descargar información no etiquetada de la Web. El criterio utilizado para seleccionar los sustantivos que se muestran en la tabla 6.1, fue que contaran con instancias de entrenamiento para todas los sentidos (categorías), ya que sin estas no se puede aplicar el método.

6.3.4 Resultados de referencia

La tabla 6.2 muestra los resultados base obtenidos utilizando dos algoritmos de categorización distintos: naïve Bayes y SVM. En todos los casos se determinó el contexto de las palabras ambiguas usando un tamaño de ventana igual a cinco (esto es, cinco palabras a la derecha y cinco palabras a la izquierda de la palabra ambigua). Además, fueron removidas las palabras de paro, signos de puntuación y símbolos numéricos.

Tabla 6.2: Resultados de referencia usando NB y SVM, SemEval

Sustantivo	Exactitud de categorización	
	NB	SVM
source	77.14	74.29
bill	92.08	95.05
president	89.20	89.20
state	78.57	78.57
management	77.27	85.82
condition	66.66	72.72
policy	74.36	87.18
rate	86.90	87.59
drug	78.26	71.74

Como se puede observar en la tabla 6.2, la exactitud de categorización se ve afectada directamente por el número de instancias de entrenamiento y el grado de desbalanceo (mostrado en la tabla 6.1). De aquí surge la idea de incrementar el tamaño del conjunto de entrenamiento para mejorar la exactitud de categorización. Este resultado muestra la necesidad de incrementar el tamaño del conjunto de entrenamiento y en este caso se hará incorporando nuevos ejemplos de entrenamiento no etiquetados descargados de la Web. Los resultados se muestran en el siguiente apartado.

6.3.5 Resultados obtenidos al aplicar el método propuesto en WSD

Este apartado describe la aplicación del método semi-supervisado propuesto a la tarea de WSD. El método, descrito en el capítulo 4, incluye dos procesos principales: la adquisición de corpus de la Web y aprendizaje semi-supervisado. A continuación se presentan los resultados obtenidos en este experimento para estos dos procesos.

La tarea central de la adquisición de corpus es descargar ejemplos no etiquetados de la Web. Estos ejemplos son descargados utilizando como patrón de búsqueda un conjunto de peticiones, las cuales son formadas por el conjunto de palabras relevantes de cada sentido. Dada la importancia que tiene el recuperar textos cortos que contengan algún contexto que pudiera ser incorporado al conjunto de entrenamiento, para este experimento en cada petición se incluyó la palabra a desambiguar, con la finalidad de que en la información recuperada se incremente la probabilidad de que dicha palabra se encuentre en el snippet. Para este experimento se consideraron sólo las primeras diez palabras relevantes por sentido, formando 120 peticiones para cada sentido, ${}_{10}C_3$. Después, usando estas peticiones, se descargaron de la Web 1,000 snippets adicionales para cada sentido de la palabra polisémica. En la tabla 6.3 se muestran, a manera de ejemplo, algunas de las peticiones construidas para el sustantivo *drug*, el cual tiene dos sentidos.

Tabla 6.3: Ejemplos de peticiones para *drug*, SemEval

Sentido	Petición
drug-1	drug new used
	drug said company
	drug sales companies
drug-2	drug trafficking charges
	drug charges major
	drug major use

Como se puede apreciar en la tabla 6.3, independientemente del número de sentidos que tenga la palabra polisémica, todas contienen la palabra *drug*.

Para este experimento, el número de ejemplos que se incorporó al conjunto de entrenamiento en cada iteración se determinó de acuerdo con la siguiente condición: la información incorporada –expresada en número de palabras– es proporcionalmente pequeña con respecto al conjunto de entrenamiento original. Esta condición es muy importante, debido al tamaño pequeño de la oración. El contexto fue cortado a únicamente cinco palabras a la izquierda y cinco palabras a la derecha de la palabra polisémica. Debido a esto, el número de palabras disminuyó significativamente y se decidió incorporar únicamente cinco ejemplos no etiquetados por sentido en cada iteración. Sin embargo, es necesario perfeccionar esta condición para poder determinar el mejor valor de *m* del algoritmo (véase el capítulo 4) para esta tarea.

La tabla 6.4 muestra los resultados de este experimento. En estos resultados se puede observar que el método permite mejorar los resultados de referencia especialmente cuando se utiliza como algoritmo de categorización naïve Bayes.

Tabla 6.4: Resultados conjunto de entrenamiento enriquecido, SemEval

Sustantivo	Bayes	It. 1	It. 2	It. 3	SVM	It. 1	It. 2	It. 3
source	77.1	80.0	80.0	80.0	74.3	77.1	80.0	68.6
bill	92.0	92.0	92.1	91.1	95.1	95.1	95.1	93.1
president	89.2	87.5	88.1	88.1	89.2	89.8	89.8	87.5
state	78.5	80.0	78.6	80.0	78.6	78.6	78.6	78.6
managment	77.2	79.5	79.5	79.5	85.8	81.8	81.8	81.8
condition	66.6	66.6	66.7	63.6	72.7	72.7	75.8	75.8
policy	74.3	76.9	76.9	74.4	87.2	87.2	87.2	74.8
rate	86.9	86.9	89.0	89.0	87.6	87.6	86.9	74.4
drug	78.2	80.4	80.4	80.4	71.7	71.7	69.6	86.9

Dada la complejidad de la tarea, se puede observar que los mejores resultados son obtenidos en las primeras dos iteraciones en la mayoría de los casos. Éste es un hecho relevante, ya que permite ver la efectividad del método semi-supervisado para la selección de las mejores instancias no etiquetadas. El proceso de selección es el siguiente: la información es descargada de la Web una sola vez, y con ella se forma una "bolsa de instancias no etiquetadas" de donde, en cada iteración, las mejores instancias son extraídas e incorporadas al conjunto de entrenamiento.

Sin embargo, después de varias iteraciones, las instancias de mayor calidad ya han sido seleccionadas, quedando sólo aquellas de calidad menor, las cuales no sólo no logran incrementar la exactitud de categorización al incorporarse al conjunto de entrenamiento sino que la disminuyen. Las iteraciones realizadas nos permiten ver la efectividad de incorporar información no etiquetada proveniente de la Web al conjunto de entrenamiento al superar el valor de exactitud de referencia obtenido con el conjunto de entrenamiento original. Estos resultados confirman la idea intuitiva de que en escenarios en los que se tienen pocos ejemplos de entrenamiento es mejor incluir un pequeño grupo de ejemplos no etiquetados, con la condición de que estos ejemplos permitan incrementar las diferencias entre los sentidos de la palabra ambigua.

6.3.6 Discusión de los resultados

Con la finalidad de comprender mejor los resultados obtenidos, en este apartado se presenta una discusión más detallada, así como las medidas adicionales implementadas, que permitirán entender mejor qué es lo que pasa al incorporar la información no etiquetada proveniente de la Web al conjunto de entrenamiento. Específicamente se presentan los resultados obtenidos al calcular la medida *SLMB* así como las medidas del vocabulario tanto en el conjunto de entrenamiento original como en el de prueba. Además se presenta el comportamiento del vocabulario (palabras distintas y total de palabras), así como las gráficas de similitud para algunos de los sustantivos utilizados, con la finalidad de tener una idea clara de la cantidad de información nueva que se incorpora al conjunto de entrenamiento.

En la tabla 6.5 se muestra el resultado de la medida *SLMB*, así como los valores del vocabulario para los conjuntos original y enriquecido en la tercera iteración. En todos los casos, se llevó a cabo la evaluación usando el conjunto de prueba proporcionado. Por un lado, se puede observar que las palabras *state*, *management*, *policy*, *bill* y *rate* no cambian significativamente su modelo de lenguaje original con respecto a la versión enriquecida del corpus.

Por lo tanto, no hubo cambios significativos en las relaciones de dependencia entre las palabras (características), lo que lleva a obtener un comportamiento similar del sistema de categorización basado en naïve Bayes con ambos corpus.

Tabla 6.5: Corpus original y enriquecido, medidas SLMB y vocabulario.

Sustantivo	Corpus original			Corpus enriquecido		
	SLMB	PD	TP	SLMB	PD	TP
bill	22.57	1306	4788	23.86	1469	5225
state	23.88	2124	7290	25.44	2261	7713
management	4.99	1168	3312	5.46	1322	3761
policy	41.27	1320	3944	43.58	1391	4151
rate	20.73	2233	12001	21.13	2296	12344
president	63.70	2594	10407	103.31	2756	10890
source	45.80	628	1803	60.41	947	2608
condition	26.14	635	1558	28.27	732	1876
drug	5.33	939	2440	0.44	1013	2616

Por otro lado, podemos ver que las palabras *president*, *condition*, *source* y *drug* han obtenido importantes cambios en los valores obtenidos de *SLMB* lo que significa que sus modelos de lenguaje se han modificado lo suficiente para conservar los mismos resultados similares con ambos corpus (original y enriquecido). Sin embargo, el sistema de categorización basado en SVM puede haberse beneficiado de este último hecho, considerando que sus vectores de soporte se han enriquecido, lo que permite que el sistema de categorización basado en SVM obtenga mejores resultados que el basado en naïve Bayes en estas últimas palabras ambiguas.

Con la finalidad de ver el impacto de la información no etiquetada incorporada al conjunto de entrenamiento, a continuación como ejemplo se presentan las gráficas de similitud para los sustantivo *rate*, *condition*, *source*. Para el sustantivo *rate* el mejor resultado se obtuvo con NB, mientras que *condition* y *source* con SVM. En la figura 6.2 se muestra la gráfica de similitud para *rate* correspondiente al conjunto de prueba, utilizando como argumentos de categorización los correspondientes al conjunto de entrenamiento original. Se utilizó un umbral de similitud igual a 0.35.

Como se puede observar en la figura 6.2, la gráfica de similitud es totalmente dispersa, la confusión es muy grande y no se puede detectar grupo alguno. En la figura 6.3 se muestra la gráfica de similitud correspondiente al conjunto de prueba de *rate*, pero ahora con los atributos de categorización correspondientes al conjunto de entrenamiento enriquecido. Se puede observar en esta figura que aunque la confusión permanece, se logra hacer una pequeña diferencia entre los grupos al poner una línea clara, aunque no exactamente a la mitad, entre los sentidos.

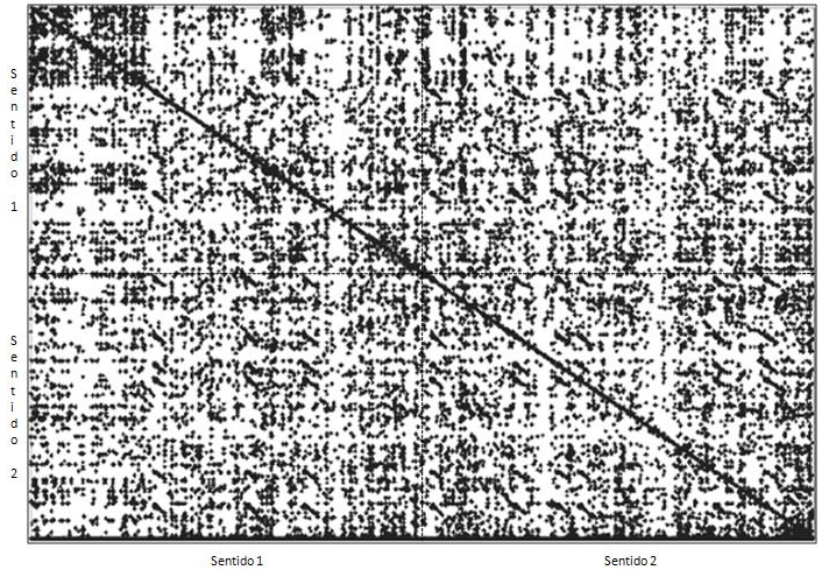

Figura 6.2: Gráfica de similitud conjunto de entrenamiento,*rate*, SemEval

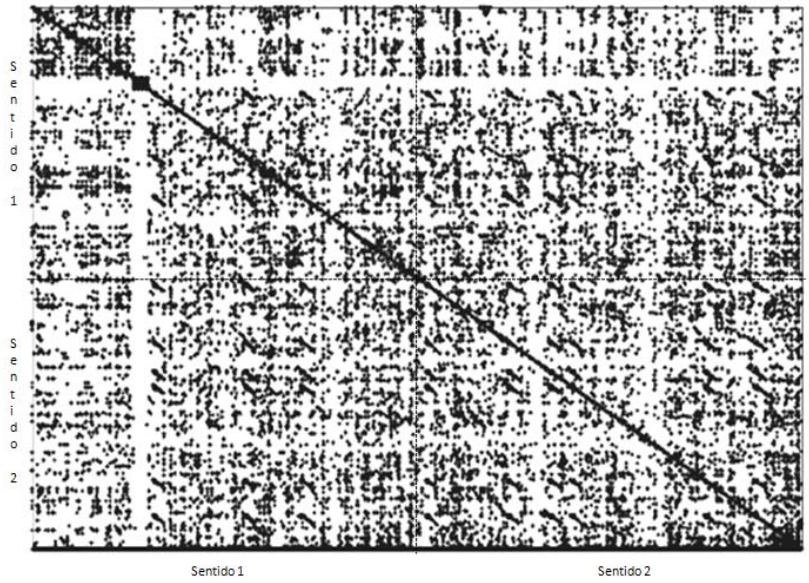

Figura 6.3: Gráfica de similitud conjunto de entrenamiento enriquecido, *rate*, SemEval

En las figuras 6.4 y 6.5 se muestran las gráficas de similitud, de la palabra *condition*, para el conjunto de entrenamiento original y el conjunto de entrenamiento enriquecido respectivamente. En ambas gráficas se utilizó un umbral de similitud de 0.5.

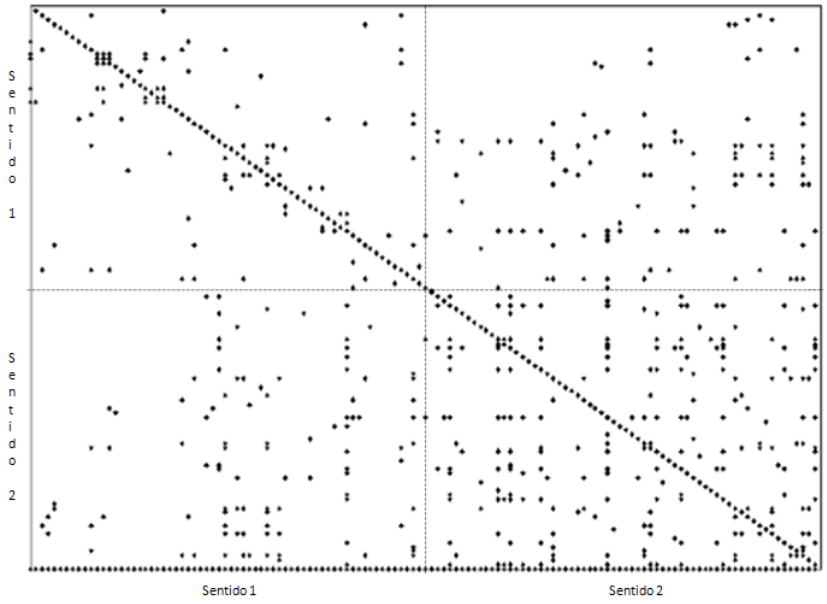

Figura 6.4: Gráfica de similitud conjunto de entrenamiento, *condition*, SemEval

Como se puede apreciar en las figuras 6.4 y 6.5, dado que todos los archivos contienen al menos a la palabra ambigua, en este caso es difícil separar los grupos ya que todos tienen algún grado de similitud mayor que cero. Sin embargo, se puede apreciar una diferencia para el sentido dos. Esta diferencia se incrementa un poco para la gráfica de similitud correspondiente al conjunto enriquecido (figura 6.5).

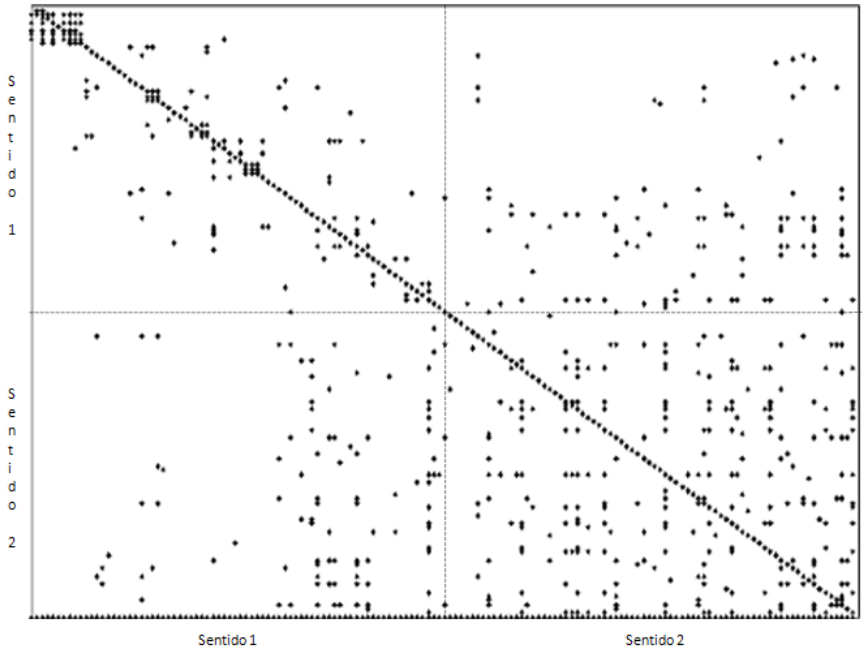

Figura 6.5: Gráfica de similitud conjunto de entrenamiento enriquecido, *condition*, SemEval

Como se puede observar estas gráficas no separan los grupos de los sentidos de una manera clara, sin embargo, se puede observar que mejoran con el conjunto enriquecido. Esto nos habla de la dificultad de esta tarea a la hora de pretender encontrar el sentido correspondiente a una palabra ambigua en función del contexto de la misma.

En la figura 6.6 se muestra la gráfica de similitud para el sustantivo *source*. En este caso el sustantivo tiene 5 sentidos. Sin embargo, se observa un alto grado de confusión entre ellos. Esta gráfica de similitud corresponde al conjunto de entrenamiento original.

En la figura 6.7 se muestra la gráfica de similitud correspondiente al sustantivo *source* utilizando como atributos de categorización los correspondientes al conjunto de entrenamiento enriquecido. Aunque se puede apreciar una mejora, sólo se pueden distinguir 4 de los 5 grupos que corresponden a los sentidos. Este hecho tiene relación con el número de instancias de entrenamiento correspondientes a cada sentido debido a que el conjunto de entrenamiento no está balanceado.

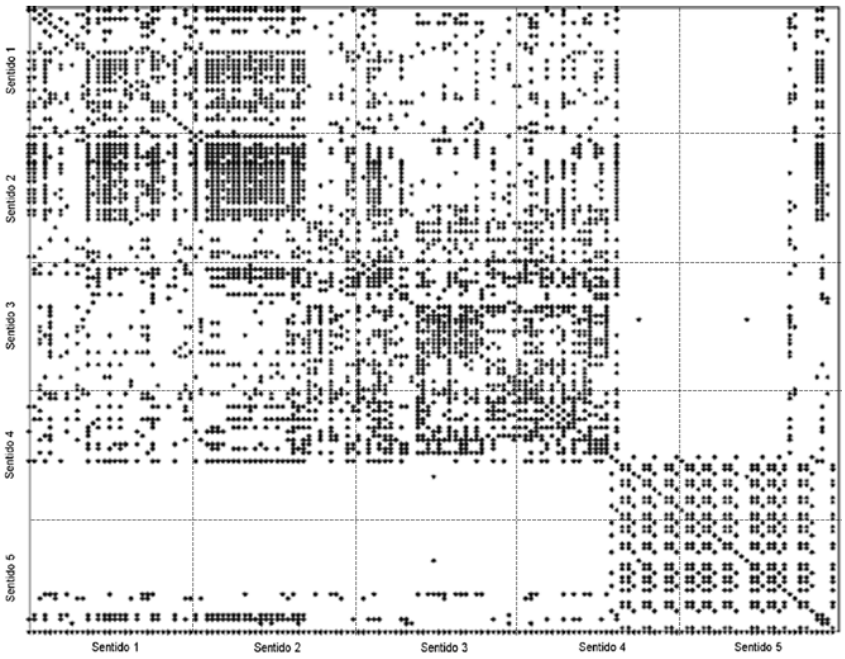

Figura 6.6: Gráfica de similitud conjunto de entrenamiento, *source*, SemEval

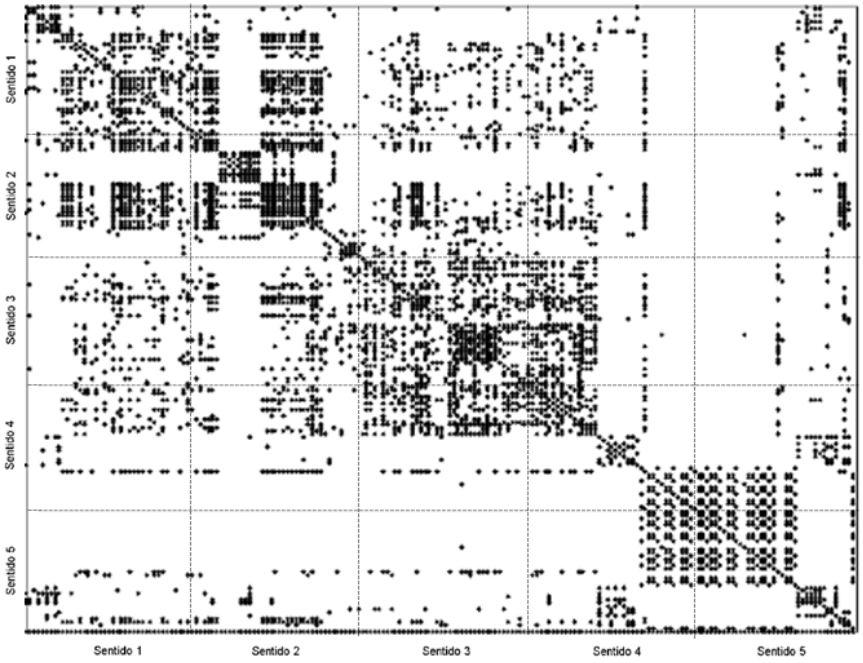

Figura 6.7: Gráfica de similitud conjunto de entrenamiento enriquecido, *source*, SemEval

6.4. Conclusiones del capítulo

En este capítulo se presentaron los resultados obtenidos al aplicar el método semi-supervisado basado en la Web a la tarea de WSD. En la primer parte del capítulo se presentó una descripción de la tarea así como los principales enfoques utilizados para su solución. Además de la descripción de la tarea se presentaron los principales enfoques de evaluación.

Dentro de los diversos enfoques utilizados para resolver el problema de WSD se presentó una clasificación de los métodos más relevantes que requieren una fuente de conocimiento externa para llevar a cabo esta tarea. Se mencionaron, además, las principales fuentes de conocimiento externa utilizadas en la tarea de WSD.

En la segunda parte de este capítulo se presentaron los resultados obtenidos al aplicar el método propuesto a la tarea de la desambiguación del sentido de las palabras. La evaluación experimental se llevó a cabo con un subconjunto de sustantivos correspondiente a la colección proporcionada en la competición SemEval. Los experimentos realizados permitieron ver la funcionalidad del método propuesto y se logró mejorar los resultados obtenidos como referencia.

Capítulo VII

Conclusiones, aportaciones y trabajo futuro

En este capítulo se presentan las conclusiones del presente trabajo de tesis obtenidas de los resultados experimentales realizados. También se exponen las principales aportaciones del método propuesto, resaltando aquellas que lo hacen diferente de otras aproximaciones existentes en el estado del arte. Además, se presentan las líneas de investigación abiertas que pueden ser desarrolladas para dar continuidad al presente trabajo de investigación. Por último, se presenta una lista de las publicaciones logradas derivadas de la investigación desarrollada.

7.1 Conclusiones

En el presente trabajo de tesis se desarrolló un nuevo método de categorización automática de documentos basado en aprendizaje semi-supervisado. Este método utiliza un pequeño conjunto de documentos etiquetados, los cuales sirven para entrenar el sistema de categorización. El conjunto de entrenamiento es enriquecido incorporando ejemplos no etiquetados. Los ejemplos no etiquetados son descargados de la Web, pasando previamente por un proceso de selección, que permite la incorporación sólo de las mejores instancias no etiquetadas al conjunto de entrenamiento. La finalidad de incorporar información no etiquetada al conjunto de entrenamiento es mejorar la exactitud de categorización.

La evaluación del método se llevó a cabo por medio de cuatro experimentos en diferentes tareas y en dos diferentes lenguajes. Tres de estos experimentos corresponden a la tarea de categorización, tanto temática como no-temática, de documentos y un experimento sobre la desambiguación del sentido de las palabras. En todos los casos se formaron conjuntos de entrenamiento y prueba, donde el conjunto de prueba no fue visto nunca por el conjunto de entrenamiento. Específicamente el método desarrollado fue evaluado en los siguientes escenarios:

(i) Categorización de noticias sobre desastres naturales (en español).
(ii) Categorización de la distribución ModApte de Reuters (en inglés)
(iii) Atribución de autoría de poemas (en español).
(iv) Desambiguación del sentido de las palabras, usando sustantivos de la colección SemEval (en inglés).

Los cuatro experimentos fueron llevados a cabo utilizando una configuración común, aplicando el mismo pre-procesamiento, así como los algoritmos de categorización y medidas de evaluación. Sin embargo, fue necesario realizar ajustes al método para que se pudiera adaptar a determinadas tareas, como es el caso que se presentó al trabajar con la colección Reuters, experimento (ii), en donde, debido al alto grado de traslape, se incorporó el modulo de sub-muestreo para balancear las categorías.

En el caso de la desambiguación del sentido de las palabras, experimento (iv), fue necesario cortar las oraciones a un tamaño de ventana igual a cinco, esto es, cinco palabras a la izquierda y cinco palabras a la derecha de la palabra ambigua. En ambos casos las adaptaciones hechas se explican en detalle en los capítulos correspondientes.

Algunas conclusiones obtenidas a partir de los experimentos realizados se mencionan a continuación:

- El método semi-supervisado de categorización automática de documentos basado en la Web funciona adecuadamente en diferentes dominios y lenguajes, como se puede desprender de los resultados obtenidos en los cuatro experimentos llevados a cabo en el presente trabajo, los cuales tienen características muy distintas ya que involucran la categorización temática, no temática y la desambiguación del sentido de las palabras, dos experimentos se realizaron en español y dos en ingles. Esta es una característica muy importante del método propuesto ya que no se limita a un idioma o dominio de aplicación.
- Se pudo observar que el método es particularmente útil cuando se tienen muy pocos ejemplos de entrenamiento, específicamente al observar los experimentos llevados a cabo en la categorización de noticias en español, experimento (i), donde se realizaron pruebas con cuatro conjuntos de entrenamiento diferentes formados por menos de diez instancias etiquetadas por clase. Esta característica es muy importante ya que muchas aplicaciones de la vida real se caracterizan por la dificultad para obtener documentos de entrenamiento, como puede ser la atribución de autoría donde generalmente se cuenta con pocos ejemplos etiquetados como entrenamiento, el contar con un sistema que puede entrenar a un sistema de categorización automática de documentos a partir de un número reducido de instancias de entrenamiento es sin duda una ventaja competitiva respecto a los sistemas de categorización existentes.
- También se observa un muy buen desempeño en el caso de la categorización de noticias en inglés de la colección de Reuters, experimento (ii), que se caracteriza por un alto grado de traslape entre clases y un marcado desbalanceo. En este caso, aprovechando que el método es adecuado para trabajar con pocos ejemplos de entrenamiento, se llevó a cabo un sub-muestreo aleatorio reduciendo el conjunto de entrenamiento de 7,220 instancias manualmente etiquetadas a sólo 100 por cada categoría, logrando superar los resultados de exactitud obtenidos al usar todo el conjunto de entrenamiento.

En los problemas que involucran categorías desbalanceadas, comúnmente la categoría con menos instancia de entrenamiento es la que interesa categorizar, como ocurren por ejemplo en la detección de fraudes, lo que nos da una clara idea de la importancia que tiene esta tarea.

- Por otro lado, también se observó un buen desempeño en el caso de la atribución de autoría (ejemplo de categorización no-temática), experimento (iii), a pesar de que es una tarea muy distinta a las anteriores, ya que en este caso la categorización se da en base al estilo de escritura, más que a las palabras que conforman el documento. El desarrollo de este experimento nos permitió comprobar que es posible descargar información de estilo de la Web, la cual puede ser incorporada al conjunto de entrenamiento con la finalidad de incrementar la exactitud de categorización.
- El método para construir las peticiones es apropiado. Los resultados obtenidos en todos los experimentos indican que es posible descargar información relevante de la Web que contribuya a mejorar la exactitud de la categorización. Este es un hecho relevante, principalmente cuando se tienen muy pocos ejemplos de entrenamiento, como en el caso de la categorización de noticias acerca de desastres naturales. También, es importante cuando se tiene el problema de la categorización por estilo, ya que en esta tarea generalmente se cuenta con pocos documentos para cada autor. En ambos casos se pudo comprobar la efectividad del método propuesto. Cuando se realiza la descarga de información de la Web se forman peticiones por categoría por medio de las palabras de mayor relevancia a la categoría utilizando las medidas de frecuencia y ganancia de información. De esta manera la información descargada es organizada en función de la categoría a la cual pertenece la petición de búsqueda, logrando con esto incrementar la probabilidad de que la información no etiquetada en realidad pertenezca a esta categoría.

Los resultados obtenidos permiten ver la efectividad del método en los distintos escenarios en los cuales se ha aplicado el método propuesto, ya que en el estado del arte cada uno de estos problemas representa una línea de investigación abierta, caracterizándose por desarrollar sistemas de categorización dedicados únicamente a la solución de una de estas problemáticas a la vez.

7.2 Aportaciones

Respecto a las aproximaciones de categorización textual que utilizan métodos semi-supervisados, el método desarrollado en la presente tesis tiene las siguientes diferencias principales:

- No requiere un conjunto previo de documentos *no* etiquetados, ya que el método propuesto descarga ejemplos no etiquetados de la Web; aunque cabe aclarar que el método funcionaria exactamente igual si se provee una colección grande de ejemplos no etiquetados. En este caso la selección de las mejores instancias se realizaría sobre el conjunto proporcionado, omitiendo la parte de descarga de información de la Web. El no requerir un conjunto previo de documentos, es una característica muy importante ya que existen aplicaciones en dominios en los que es muy difícil contar con documentos que puedan ser utilizados para incorporarse al conjunto de entrenamiento con la finalidad de incrementar la exactitud de categorización en un sistema de categorización.

  Por ejemplo, cuando se tiene un problema de atribución de autoría, donde generalmente sólo existen pocos documentos de cada autor y a partir de ellos se debe llevar a cabo la solución del problema. En este caso, el método propuesto no encuentra exactamente esos documentos particulares en la Web, pero localiza fragmentos de texto, ejemplos no etiquetados, que tienen una distribución de palabras similar y entonces pueden ser consideradas como instancias adicionales de entrenamiento.
- El método desarrollado aplica una aproximación basada en aprendizaje semi-supervisado para seleccionar las mejores instancias no etiquetadas descargadas de la Web. Para llevar a cabo esta tarea no sólo se considera la categoría asignada a cada instancia por el sistema de categorización, también se considera una categoría previa que le es asignada al momento de realizar la descarga de la Web, esto es, una categoría a priori asignada al momento de formar las peticiones que serán lanzadas a la Web para descargar los ejemplos no etiquetados.

El método propuesto utiliza esta información adicional para predecir la categoría de un ejemplo no etiquetado. Este proceso depende del número de instancias etiquetadas que se tengan y, por consecuencia, resulta muy adecuado para realizar trabajos de categorización cuando se tienen muy pocos ejemplos de entrenamiento.

7.3 Trabajo futuro

Como parte del trabajo futuro, que permita por un lado mejorar el método desarrollado, y por otro lado, ampliar su aplicación a otras áreas, existen varias actividades que se pueden realizar, entre la que destacan:

- En los experimentos realizados en el presente trabajo de tesis, el número de ejemplos no etiquetados que se incorporaron al conjunto de entrenamiento se determinó de acuerdo con la siguiente condición: la información incorporada, expresada en número de palabras, debía ser proporcionalmente pequeña con respecto al conjunto de entrenamiento original. Sin embargo es necesario que en futuros experimentos se realice un análisis mayor con la finalidad de mejorar este aspecto. Específicamente se deben tener criterios claros para determinar los valores de m y σ del algoritmo propuesto, en donde m representa los mejores ejemplos de cada una de las categorías que serán incorporadas al conjunto de entrenamiento en cada iteración, mientras que σ es el número de iteraciones realizadas.
- La aplicación del método propuesto en otras tareas del procesamiento del lenguaje natural que requieran de categorización tales como la categorización de entidades nombradas que podríamos considerar como una categorización temática, ya que existen reglas que permiten identificar los diferentes tipos de entidades. En este caso se podría pensar en utilizar la Web como fuente de ejemplos de entidades nombradas: contar con conjuntos de entrenamiento suficientemente grandes para que los sistemas basados en aprendizaje automático puedan realizar su tarea de una manera adecuada, es uno de los mayores problemas en esta tarea.

- Adicional al punto anterior, también sería interesante aplicar el método desarrollado en la categorización de correos electrónicos, donde no hay una clara división de si se trata de un problema de categorización temático o de estilo y sería interesante ver el comportamiento del método presentado. Otro campo donde aplicar el método propuesto es la categorización de opiniones. En este caso se aborda nuevamente una categorización no temática (se está de acuerdo o en desacuerdo a cierto suceso) donde los términos subjetivos dan pie a la categorización del documento.

7.4 Publicaciones

En este apartado se presentan las publicaciones obtenidas con el trabajo de investigación presentado en esta tesis, las cuales se enumeran a continuación:

1. USING THE WEB AS CORPUS FOR SELF TRAINING TEXT CATEGORIZATION.
   Journal of information retrieval, vol. 12, issue 3. Special Issue on Non-English Web Retrieval, pp. 400-416, issn: 1386-4564, Journal no. 10791, año: 2009
   Rafael Guzmán Cabrera, Manuel Montes y Gómez, Paolo Rosso y Luis Villaseñor Pineda

2. SEMI-SUPERVISED WORD SENSE DESAMBIGUATION USING THE WEB AS CORPUS
   LNCS 5449, Springer Verlag, issn 0302-9743, CICLING-2009
   Rafael Guzmán Cabrera, Paolo Rosso, Manuel Montes y Gómez, Luis Villaseñor Pineda y David Pinto Avendaño

3. A WEB BASED SELF-TRAINING APPROACH FOR AUTHORSHIP ATTRIBUTION
   LNAI 5221, Springer Verlag, issn: 0302-9743, GOTAL-2008
   Rafael Guzmán Cabrera, Manuel Montes y Gómez, Paolo Rosso y Luis Villaseñor Pineda

4. TAKING ADVANTAJE OF THE WEB FOR TEXT CLASSIFICATION WITH IMBALANCED CLASSES
   LNAI 4827, Springer Verlag, issn 0302-9743; MICAI-2007
   Rafael Guzmán Cabrera, Manuel Montes y Gómez, Paolo Rosso y Luis Villaseñor Pineda

5. IMPROVING TEXT CLASSIFICATION BY WEB CORPORA
   Advances in soft computing, Springer Verlag, Web Intelligence Conference 2007
   Rafael Guzmán Cabrera, Manuel Montes y Gómez, Paolo Rosso y Luis Villaseñor Pineda

6. SEARCHING THE WEB FOR SIGNIFICANT WORD SENSE COLLOCATIONS
   IEEE-MEP-2006, Mexico, isbn: 1-4244-0627-7
   Rafael Guzmán Cabrera, Manuel Montes y Gómez y Paolo Rosso

7. BÚSQUEDA DE COLOCACIONES EN LA WEB PARA SINÓNIMOS DE WORDNET
   Revista Acta universitaria, issn 0188-6266, vol. 15, No. 2, año:2005
   Rafael Guzmán Cabrera, Manuel Montes y Gómez y Paolo Rosso

8. MINING THE WEB FOR SENSE DISCRIMINATION PATTERNS
   ieee- ictis 2005, Marruecos, isbn: 9954-8577-0-2
   Rafael Guzmán Cabrera, Paolo Rosso, Manuel Montes y Gómez y José Manuel Gómez Soriano

9. SEARCHING THE WEB FOR WORD SENSE COLLOCATIONS
   electro-2005, México, isbn:1405-2172
   Rafael Guzmán Cabrera, Manuel Montes y Gómez y Paolo Rosso

Referencias

(Aas et al., 1999) Aas K. and Eikvil L., *Text categorization: A survey*. Technical Report, Norwegian Computing Center, 1999.

(Agirre et al., 1996) Agirre E. and Rigau G., *A Proposal for Word Sense Disambiguation using Conceptual Distance*, Proceedings recents advances in natural language processing, 1996.

(Agirre et al., 2006) Agirre, Eneko & Philip Edmonds (eds.), *Word Sense Disambiguation Algorithms and Applications,* Springer, 2006.

(Agirre et al., 2007) Agirre E., Magnini B., Lopez O., Otegi A., Rigau G. and Vossen P., *SemEval-2007 Task01: Evaluating WSD on Cross-Language Information Retrieval*, Proceedings of CLEF 2007 Workshop, 2007.

(Alcaraz et al., 1997) Alcaraz E. y Martínez A., *Diccionario de Lingüística Moderna*. Editorial Ariel, S.A., 1997.

(Argamon et al., 2003) Argamon S. and Sterling S., *Learning Algorithms and Features for Multiple Authorship Discrimination*. IJCAI'03 Proceedings, Workshop on Computational Approaches to Style Analysis and Synthesis, 2003.

(Argamon et al., 2005) Argamon S. and Levitan S., *Measuring the Usefulness of Function Words for Authorship Attribution*. Proceedings of Association for Literary and Linguistic Computing Association Computer Humanities, 2005.

(Apte et al., 1994) Apte C., Damerau F., Weiss S. M., *Automated learning of decision rules for text categorization*, ACM transaction of information retrieval systems, Vol 12, pp: 233-251, 1994.

(Aurajo, 2006) Aurajo B. S., *Aprendizaje Automático: Conceptos básicos y avanzados*, Pearson-Prentice Hall, 2006.

(Banea et al., 2008) Banea C., Mihalcea R., Wiebe J. and Hassan S., *Multilingual Subjectivity Analysis Using Machine Translation*, Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2008.

(Baroni et al., 2004) Baroni M. and Bernardini S., *BootCat: Boostrapping corpora and terms from the web*, Proceedings LREC, 2004.

(Barron et al., 2009) Barron-Cedeño A. and Rosso P., *On Automatic Plagiarism Detection Based on n-Grams Comparison*, LNCS 5478, pp: 696–700, Springer-Verlag, 2009.

(Batista et al., 2004) Batista, G., Prati, R. and Monard, M., *A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data*, SIGKDD Explorations 6(1) pp: 20–29, 2004.

(Belking et al., 2004) Belkin M., Matveeva I. and Niyogi P., *Regularization and semi-supervised learning on large graphs*, Proceedings COLT, 2004.

(Bennett et al., 1999) Bennett K. P. and Demiriz A., *Semi-supervised support vector machine*, Proceedings NIPS, 1999.

(Bennett et al., 2005) Bennett P. N., Dumais S. T. and Horvitz E., *The combination of text classifiers using reliability indicators*, Information Retrieval, 8:67–100, 2005.

(Blum et al., 1998) Blum A. and Mitchell T., *Combining labeled and unlabeled data with co-training*, Proceedings of the Workshop on Computational Learning Theory, Morgan Kaufmann Publishers, pp: 92-100, 1998.

(Blum et al., 2001) Blum A. and Chawla S., *Learning from labeled and unlabeled data using grap mincuts*, Proceedings ICML, 2001.

(Brank et al., 2003) Brank J., Grobelnik M., Milic-Frayling N., and Madenic D., *Training text classifiers with SVM on very few positive examples*. Technical Report MSR-TR-2003-34, Microsoft Research, 2003.

(Brants et al., 2006) Brants, T. and Franz, A., *Web 1t 5-gram, ver. 1*, Linguistic Data Consortium, 2006.

(Breiman, 1994) Breiman L., *Bagging predictors*, Technical Report 421, department of Statistics, Univesity of California at Berkeley, 1994.

(Breiman, 1996) Breiman L., *Bagging predictors*, Machine Learning, 34(2), pp: 123-140, 1996.

(Burges, 1998) Burges C., *A tutorial on support vector machines for pattern recognition*, Data Mining and Knowledge Discovery, 2 (2), pp: 121–167, 1998.

(Cardoso et al., 2006) Cardoso-Cachopo A. and Oliveira A. L., *Empirical evaluation of centroid-based models for single-label text categorization*, Technical Report 7/2006, INESC-ID, Lisboa, Portugal, 2006.

(Cavagliá et al., 2001) Cavagliá G. and Kilgarriff A., *Corpora from the Web*, Procedings 4th Annual CLUCK Colloquium, pp: 120-124, 2001.

(Chakrabarti et al., 1999) Chakrabarti S., Van der B. and Dom M., *Focused crawling: a new approach to topic-specific web resource discovery*, Proceedings Of the 8th int. WWW conference, pp: 545-562, 1999.

(Chan et al., 1998) Chan P.K. and Stolfo S.J., *Toward scalable learning with non-uniform class and cost distributions: a case study in credit card fraud detection*, Proceedings of the fourth ICKDDM, pp: 164-168, 1998

.

(Chapelle et al., 2005) Chapelle O. and Zien A., *Semi-Supervised classification by low density separation,* 10th workshop on AI and stat, 2005.

(Chapelle et al., 2006) Chapelle O., Scholkopf B. and Zien A. *Semi-Supervised Learning*, The MIT Press, 2006.

(Charniak et al., 2000) Charniak E., Blaheta D., Hall N., Hale K., and Johnson M., *WSJ corpus release 1,* Technical Report. *LDC2000T43*, Linguistic Data Consortium pp: 1987-1989, 2000.

(Chaski, 2005) Chaski C., *Who's at the Keyword? Authorship Attribution in Digital Evidence Investigations*, International Journal of Digital Evidence, 2005.

(Chawla et al., 2004) Chawla N., Japkowicz N. and Kotcz A., *Editorial: Special Issue on Learning from Imbalanced Data Sets,* kdd Explorations Volume 6, Issue 1, pp: 1-6, 2004.

(Chen et al., 2007) Chen K. and Wang S., *Regularized boost for semi-supervised learning*, Proceedings NIPS, 2007.

(Clear, 1993) Clear J. *The British National Corpus*, Digital Word: Text-Based Computing in the Humanities, MIT Press, pp: 163–187, 1993.

(Cohen et al., 2004) Cohen I., Cozman F., Sebe N., Cirelo M. and Huang T., *Semi-supervised learning of classifiers with application to human-computer interaction*, IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 26, 2004.

(Corney et al., 2002) Corney M., de Vel O., Anderson A. and Mohay G., *Gender-Preferential Text Mining of E-mail Discourse*, 18th Annual Computer Security Applications Conference, pp: 9-13, 2002.

(d'Alche Buc et al., 2006) d´Alche Buc F., Grandvalet Y. and Ambroise C., *Semi-supervised margin-boots*, Proceedings NIPS, 2002

(de Vel O et al.,, 2001) de Vel O., Anderson A., Corney M. and Mohay G., *Mining Email Content for Author Identification Forensics*, Special Section on Data Mining for Intrusion Detection and Threat Analysis SIGMOD Record, volume 30 Issue 4, pp: 55-64, 2001.

(Debole et al., 2003) Debole, F. and Sebastiani F., *Supervised term weighting for automated text categorization*, Proceedings ACM Symposium on Applied Computing, pp: 784-788, 2003.

(Diederich et al., 2003) Diederich J., Kindermann J., Leopold E. and Paas G., *Authorship Attribution with Support Vector Machines*, Applied Intelligence. Kluwer Academic Pushishers, volume 19 Issue 1, pp: 109-123, 2003.

(Dietterich, 1997) Dietterich G.T., *Machine learning research: four current directions*, AI magazine volume 18, Issue 4, pp: 97-136, 1997.

(Dietterich, 2000) Dietterich G.T., *Ensenble methods in machine learning*, *First* workshop on multiple classifier systems. Springer Verlag, 1857, pp: 1-15, 2000.

(Di Nunzio, 2009) Di Nunzio Giorgio M., *Using scatterplots to understand and improve probabilistic models for text categorization and retrieval*, International Journal of Approximate Reasoning, Elsevier, volume 50, Issue 4, pp: 581-594, 2009.

(Domingos, 1999) Domingos, P., *Metacost: A General Method for Making Classifiers Cost-sensitive*, Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp: 155–164, 1999.

(Duda et al., 1973) Duda R. O. and Hart P. E., *Pattern Classification and Scene Analysis*, Wiley, New York, 1973.

(Duffield et al., 2007) Duffield, C. J., Hwang, J. D., Brown, S. W., Dligach, D., Vieweg, S. E., Davis, J. and Palmer, M., *Criteria for the manual grouping of verb senses*, Proceedings of the linguistic Annotation Workshop, 2007.

(Dzeroski et al., 2000) Dzeroski S. and Zenki B., *Is combining classifiers better than selecting the best one*, ICML, pp: 123-130, 2000.

(Edmonds, 2000) Edmonds P., *Designing a task for SENSEVAL-2*, Technical note, University of Brighton, U.K., 2000.

(Eibe et al., 2006) Eibe F., Remco R. and Bouckaert N., *Naive Bayes for Text Classification with Unbalanced Classes*, LNCS, Springer-Verlag, volume 4213, pp: 503-510, 2006.

(Eibl et al., 2005) Eibl G. and Pfeiffer K.P., *Multiclass-boosting for weak classifier*, Journal of Machine Learning Research, 6, pp: 189-210, 2005.

(Fellbaum, 1998) Fellbaum C., *WordNet: An electronic Database*, MIT Press, Cambridge, 1998.

(Ferrández et al., 2008) Ferrández, O., Muñoz, R. and Palomar, M., *Improving Question Answering Tasks by Textual Entailment Recognition*, LNCS, Springer-Verlag, volume 5039, pp: 339-340, 2008.

(Ferraresi et al., 2008) Ferraresi A., Zanchtta E., Bernardini S. and Baroni M., *Introducing and evaluating ukWaC, a very large web-derived corpus of English*, Proceedings, 4th WAC workshop, 2008.

(Finn et al., 2003) Finn A. and Kushmerick N., *Learning to classify documents according to genre*, Proceedings, Workshop on Computational Approaches to Style Analysis and Synthesis, 2003.

(Frakes et al., 1992) Frakes, W. and Baeza, R. *Information retrieval: data structures and algorithms*, Prentice Hall, London, 1992.

(Freund et al., 1996) Freund Y. and Schapire R., *Experiments with a new boosting algorithm*, Proceedings, International Conference on Machine Learning, pp: 148-156, 1996.

(Fürnkranz, 1998) Fürnkranz J., *A Study Using n-gram Features for Text Categorization*, Technical Report OEFAI-TR-9830, Austrian Institute for Artificial Intelligence, 1998.

(Gelbukh et al., 2001) Gelbukh A. and Sidorov G., *Zipf and Heaps Laws' Coefficients Depend on Language*, Proceedings, and Conference on Intelligent Text Processing and Computational Linguistics, LNCS, Springer-Verlag, pp: 332–335, 2001.

(Glenn et al., 2006) Glenn J. and Myatt N., *Making Sense of Data: A Practical Guide to Exploratory Data Analysis and Data Mining*, John Wiley, 2006.

(Govert, 1999) Govert N. A., *Probabilistic description-oriented approach for categorizing web documents*, 1999.

(Graff, 2003) Graff D., *English gigaword*, Technical report, Linguistic Data Consortium, 2003.

(Graham et al., 2005) Graham, N., Hirst, G. and Marthi, B., *Segmenting documents by stylistic character*, Natural Language Engineering, volume 11, issue 4, pp: 39-75, 2005.

(Grieve, 2007) Grieve, J., *Quantitative Authorship Attribution: An Evaluation of Techniques*, Literary and Linguistic Computing, volume 22, issue 3, pp: 251-270, 2007.

(Gruber, 1993) Gruber, T. R., *Toward principles for the design of ontologies used for knowledge sharing*, Proceedings of the International Workshop on Formal Ontology, 1993.

(Grzymala-Buse et al., 2000) Grzymala-Buse J.W., Zheng Z., Goowin L. K. and Grzymala-Buse W. J., *An approach to imbalanced data sets based on changing rule strength*. Workshop AAAI, pp: 69-74, 2000.

(Guzman et al., 2009) Guzmán-Cabrera R., Montes-y-Gomes M., Rosso P. and L. Villaseñor-Pineda, *Using the web as corpus for self training text categorization*, journal of information retrieval, Journal no. 10791, 2009.

(Guzman et al., 2009b) Guzmán-Cabrera R., Montes-y-Gomes M., Rosso P., Villaseñor-Pineda L. and Pinto-Avendaño D., *Semisupervised word sense disambiguation using the web as corpus*, LNCS 5449, Springer Verlag, 2009.

(Guzman et al., 2008) Guzmán-Cabrera R., Montes-y-Gomes M., Rosso P. and L. Villaseñor-Pineda, *A web based self training approach for authorship attribution*, LNAI 5221, Springer Verlag, 2008.

(Guzman et al., 2007) Guzmán-Cabrera R., Montes-y-Gomes M., Rosso P. and L. Villaseñor-Pineda, *Improving text classification by web corpora*, Advances in soft computing, Springer Verlag, 2007.

(Guzman et al., 2007b) Guzmán-Cabrera R., Montes-y-Gomes M., Rosso P. and L. Villaseñor-Pineda, *Taking advantage of the web for text classification with imbalanced classes*, LNAI 4827, Springer Verlag, 2007.

(Hartley et al., 1968) Hartley H. O. and Rao J. N., *Classification and estimation in analysis of variance problems*, Review of International Statistical Institute, volume 36, pp: 141-147, 1968.

(Ide, 2006) Ide N., *Making senses: Bootstrapping sense-tagged lists of semantically-related words*, Computational Linguistics and Intelligent Text, LNCS, volume 878. Springer Verlag, pp: 13–27, 2006.

(Japkowicz et al., 2002) Japkowicz N. and Shaju S., *The Class Imbalance Problem: A Systematic Study Intelligent Data Analysis*, Journal, volume 6, issue 5, pp: 1-32, 2002.

(Jin et al., 2007) Jin R., Zhang J., *Multi-class learning by smoothed boosting*, Machine learning, 67(3), pp: 207-227, 2007.

(Joachims, 1998) Joachims T., *Text categorization with support vector machines: Learning with many relevant features,* Proceedings, Tenth European Conference on Machine Learning, LNCS, volume 1398, pp: 137-142, 1998.

(Joachims, 1999) Joachims, T., *Transductive inference for text classification using support vector machines*, Morgan Kaufmann Publishers Inc., pp: 200–209,1999.

(Joachims, 2001) Joachims T., *A statistical learning model of text classification with support vector machines*, Proceedings, 24th ACM International Conference on Research and Development in Information Retrieval, ACM Press, 2001.

(Joachims, 2002) Joachims, T., *Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms*, Kluwer Academic Publishers, 2002.

(Jurafsky et al., 2000) Jurafsky, D. and Martin J. H., *Speech and Language Processing: An Introduction to Natural Language Processing,* Computational Linguistics and Speech Recognition, 1 ed., Prentice Hall, 2000.

(Keselj et al., 2003) Keselj V., Peng F., Cercone N. and Thomas C., *N-gram-based Author Profiles for Authorship Attribution*, Proceedings, Conference Pacific Association for Computational Linguistics, pp: 255-264, 2003.

(Kilgarriff et al., 2003) Kilgarriff A. and Greffenstette G., *Introduction to the Special Issue on Web as Corpus*, Computational Linguistics, volume 29, issue 3, pp: 1-15, 2003.

(Kilgarriff et al., 2000) Kilgarriff A., and Palmer M., *Introduction to the special issue on Senseval*, Computation Human, volume 34, pp: 1-13, 2000.

(Kilgarriff, 1998) Kilgarriff A., *Senseval: An exercise in evaluating word sense disambiguation programs*, Proccedings, LREC, 1998.

(Kohavi, 1995) Kohavi R., *A study of cross-validation and bootstrap for accuracy estimation and model selection*, Proccedings, IJCAI, pp: 1137-1145, 1995.

(Koller et al., 1997) Koller, D. and Sahami M., *Hierarchically classifying documents using very few words*, Proceedings, 14th International Conference on Machine Learning, pp: 170-178, 1997.

(Koppel et al., 2002) Koppel M., Argamon S. and Shimoni A., *Automatically categorizing written texts by author gender*, Literary and Linguistic Computing, volume 17, issue 4, pp: 401-412, 2002.

(Koppel et al., 2004) Koppel, M. and Schler, J., *Authorship Verification as a One Class Classification Problem*, Proceeding, ECML, 2004.

(Kotsiantis et al., 2004) Kotsiantis S. and Pintelas P., *Selective voting*, IEEE-ISDA, pp: 397-402, 2004.

(Kubat et al., 1997) Kubat M. and Matwin S., *Addressing the Curse of Imbalanced Training Sets: One Sided Selection*, Proceedings, Fourteenth International Conference on Machine Learning, pp: 179−186, Morgan Kaufmann, 1997.

(Kubat et al., 1998) Kubat M., Holte R.C. and Matwin S., *Machine learning for the detection of oil spills in satellite radar images*, Machine Learning 30(2), pp: 195-215, 1998.

(Kucera et al., 1967) Kucera H. and Francis W., *Computational Analysis of Present-Day American English*, Brown University Press, 1967.

(Lee et al., 2006) Lee C. and Lee G., *Information gain and divergence-based feature selection for machine learning-based text categorization*, Information Processing and Management: an International Journal, volume 42, issue 1, pp: 155-165, 2006.

(Lesk, 1986) Lesk Michael E., *Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from a Ice-cream Cone*, Proceedings, *ACM SIGDOC Conference*, pp: 24-26, 1986.

(Lewis et al., 1994) Lewis D. and Ringuette M., *A comparison of two learning algorithms for text classification*, Proceedings, 3rd Annual Symposium on Document Analysis and Information Retrieval, pp: 81-93, 1994.

(Lewis, 1998) Lewis D., *Naive (Bayes) at forty: The independence assumption in information retrieval*, Proceedings, 10th European Conference on Machine Learning, Springer Verlag, pp: 4-15, 1998.

(Li, 2006) Li, L., *Multiclass boosting with repartitioning*, ICML, 2006.

(Liu et al., 2005) Liu T.Y., Yang Y., Wan H., Zhou Q., Gao B., Zeng H.J., Chen Z. and Ma W.Y., *An experimental study on large-scale web categorization*, Proceedings, 14th ACM International World Wide Web Conference, pp: 1106–1107, 2005.

(Mallapragada et al., 2007) Mallapragada P.K., Jin R., Jain A.K., Liu Y., *Semiboost: Boosting for bsemisupervised learning*, Technical report, Department of computer science and engineering, Michigan State University, 2007.

(Maloof, 2003) Maloof M. A., *Learning when data sets are imbalanced and when costs are unequal and unknown*, Workshop on learning from imbalanced data sets II, ICML, 2003.

(Marquez et al., 2006) Marquez, L. Escudero, G. Martinez D. and Rigau, G., *Supervised corpus-based methods for WSD*, Word sense disambiguation: Algorithms and applications, Springer Verlag, pp: 167-216, 2006.

(Meyer et al., 2007) Meyer Z., Stein B. and Kulig M., *Plagiarism detection without reference collections*, Proceedings, Advances in Data Analysis, pp: 359-366, 2007.

(Miller et al., 1990) Miller G. A., Beckwith R., Fellbaum C., Gross D., and Miller K., *WordNet: An online lexical database*, International Journal Lexicograph, volume 3, issue 4, pp: 235–244, 1990.

(Miller et al., 1993) Miller G. A., Leacock C., Tengi R. and Bunker R.T., *A semantic concordance*, Proceedings, ARPA Workshop on Human Language Technology, pp: 303-308, 1993.

(Mihalcea, 2004) Mihalcea R., *Co-training and self-training for word sense disambiguation*, Proceedings, 8th Conference on Computational Natural Language Learning, pp: 33-40, 2004.

(Mihalcea et al., 2006) Mihalcea R. and Liu H., *A corpus-based approach to finding happiness*, Proceedings, AAAI Spring Symposium on Computational Approaches to Weblogs, 2006.

(Mitchell, 1997) Mitchell T., *Machine Learning*, McGraw-Hill, 1997.

(Molina, 2004) Molina A., *Desambiguación en procesamiento del lenguaje natural mediante técnicas de aprendizaje automático*, Tesis Doctoral, Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, 2004.

(Molla et al., 2007) Molla D. and Vicedo-González J. L., *Question Answering in Restricted Domains: An Overview*, Computational Linguistics, volume 33, issue 1, pp: 41-61, 2007.

(Montejo et al., 2007) Montejo, A., Ureña L.A., García-Cumbreras M.A. and Perea J.M., *Using Linguistic Information as Features for Text Categorization*, Proceedings, NATO Advanced, pp: 107-108, 2007.

(Montoyo, 2002) Montoyo A., *Desambiguación léxica mediante marcas de especificidad*, Tesis Doctoral. Departamento de Lenguajes y sistemas informáticos, Universidad de Alicante, 2002.

(Navagli, 2009) Navagli R., *Word Sense Disambiguation: A Survey*, ACM computing Surveys, volume 41, issue 2, 2009.

(Nigam et al., 2000) Nigam K., McCallum A., Thrun S. and Mitchell T., *Text classification from labeled and unlabeled documents using EM*, Machine Learning, volume 39, pp: 103-114, 2000.

(Quinlan, 1996) Quinlan J.R., *Bagging, Boosting and C.45*, Proceedings, Fourteenth National Conference on Artificial Intelligence, 1996.

(Patwardhan et al., 2007) Patwardhan, S. and Riloff, E., *Effective Information Extraction with Semantic Affinity Patterns and Relevant Regions*, Proceedings, Conference on Empirical Methods in Natural Language Processing, 2007.

(Pease et al., 2002) Pease A., Niles I. and Li J., *The suggested upper merged ontology: A large ontology for the semantic Web and its applications*, Proceedings, AAAI-2002 Workshop on Ontologies and the Semantic Web, 2002.

(Pedersen, 2006) Pedersen T., *Unsupervised corpus-based methods for WSD*, Word Sense Disambiguation: Algorithms and Applications, Springer Verlag, pp: 133-166, 2006.

(Peng et al., 2004) Peng F., Schuurmans D. and Wang S., *Augmenting Naïve Bayes Classifiers with Statistical Languages Models*, Information Retrieval, volume 7, Issue 3, pp: 317-345, 2004.

(Pianta et al., 2002) Pianta E., Bentivogli L. and Girardi C., *Multi-WordNet: Developing an aligned multilingual database*, Proceedings, 1st International Conference on Global WordNet, pp: 21–25, 2002.

(Pinto, 2008) Pinto D., *On Clustering and Evaluation of Narrow Domain Short-Text Corpora*, Tesis Doctoral, Departamento de sistemas informaticos y computacion, Universidad Politécnica de Valencia, 2008.

(Philpot et al., 2005) Philpot A., Hovy E., and Pantel P., *The Omega Ontology*, Proceedings, IJCNLP Workshop on Ontologies and Lexical Resources, pp: 59–66, 2005.

(Porter et al., 1980) Porter M.F., Rijsbergen V.C. and Robertson S.E., *New models in probabilistic information retrieval*, British Library Research and Development Report, no. 5587, 1980.

(Proctor, 1978) Proctor, P., *Longman Dictionary of Contemporary English*, Longman Group, Harlow, U.K., 1978.

(Resnik et al., 1997) Resnik P. and Yarowsky D., *A perspective on word sense disambiguation methods and their evaluation*, Tagging Text with Lexical Semantics: Why, What and How? ACL SIGLEX, 1997.

(Resnik et al., 2000) Resnik P. and Yarowsky D., *Distinguishing Systems and Distinguishing Senses: New Evaluation Methods for Word Sense Disambiguation*, Natural Language Engineering volume 5, issue 2, pp: 113-133, 2000.

(Resnik et al., 2005) Resnik P. and Elkiss A., *The Linguist's Search Engine: An Overview*, Proceedings, ACL, 2005.

(Riloff et al., 2005) Riloff, E., Wiebe, J. and Phillips, W., *Exploiting Subjectivity Classification to Improve Information Extraction*, Proceedings, 20th National Conference on Artificial Intelligence, 2005.

(Roget, 1977) ROGET, P. M., *Roget's International Thesaurus*, 1st ed. Cromwell, New York, NY. 1977.

(Rosso et al., 2003) Rosso P., Masulli F., Buscaldi D., Pla F. and Molina A., *Automatic noun sense disambiguation*, Proceedings, CICLing 2003, LNCS, Springer-Verlag, pp: 275-278, 2003.

(Rosso et al., 2005) Rosso P., Montes y Gomez M., Buscaldi D., *Two web-based approaches for noun sense disambiguation*, Proceedings, CICLing 2005, Springer-Verlag, pp: 261-273, 2005.

(Salzberg, 1999) Salzberg S., *On Comparing Classifiers: A critique of Current Research and Methods*, Data Mining and Knowledge Discovery, volume 1, pp: 1-12, 1999.

(Salton, 1983) Salton G., *Introduction to modern information retrieval*. McGraw-Hill, 1983.

(Salton et al., 1987) Salton, G., and C. Buckley, *Term weighting approaches in automatic text retrieval*, Technical Report TR87-881, Cornell University, Ithaca, NY, USA, 1987.

(Sanchez, 2005) Sanchez R. R., *Selección de atributos mediante proyecciones*, tesis doctoral, Universidad de Sevilla, 2005.

(Scholkopf et al., 2003) Scholkopf, B., and Smola A., *A short introduction to learning with kernels*, Proceedings, LNAI, pp: 41–64, 2003.

(Schütze, 1998) Schütze H., *Automatic word sense discrimination*, Computational Linguistic, volume 24, issue 1, pp: 97-124, 1998.

(Sebastiani, 2002) Sebastiani F., *Machine learning in automated text categorization*, ACM Computing Surveys, volume 34, issue 1, pp: 1-47, 2002.

(Sebastiani, 2005) Sebastiani, F., *Text Categorization*, Proceedings, CRM and Knowledge Management, pp: 109-129. WIT Press, 2005.

(Sebastiani, 2006) Sebastiani F., *Classiffication of text, automatic*, The Encyclopedia of Language and Linguistics, volume 2, pp: 457-463, Elsevier, Science Publishers, 2006.

(Shannon et al., 1963) Shannon, C. E., and W. Weaver, *A Mathematical Theory of Communication*, University of Illinois Press, Champaign, IL, USA, 1963.

(Schapire et al., 2000) Schapire R. E. and Singer Y., *BoosTexter: a boosting-based system for text categorization*, Machine Learning, volume 39, issue 2., pp: 135-168, 2000.

(Schütze, 1998) Schütze, H., *Automatic word sense discrimination*, Computational Linguistic, volume 24, issue 1, pp: 97-124, 1998.

(Smucker et al., 2007) Smucker M., Allan J. and Carterette B., *A comparison of statistical significance tests for information retrieval evaluation*, Proceedings, ACM Sixteenth Conference on Information and Knowledge Management, pp: 623-632, 2007.

(Soanes et al., 2003) Soanes C., and Stevenson A., *Oxford Dictionary of English*. Oxford University Press, 2003.

(Stamatatos et al., 2000) Stamatatos E., Fakotakis N. and Kokkinakis G., *Text Genre Detection Using Common Word Frequencies*, Proceedings, 18th Internaional Conference on Computational Linguistics, pp: 808-814, 2000.

(Stamatatos, 2009a) Stamatatos E. A, *Survey of Modern Authorship Attribution Methods*, Journal of the American Society for information Science and Technology, 60(3): 538-556, 2009.

(Stamatatos, 2009b) Stamatatos, E., *Intrinsic Plagiarism Detection Using Character n-gram Profiles*, Proceedings of the SEPLN Workshop on Uncovering Plagiarism, Authorship, and Social Software Misuse, PAN-09. 2009.

(Suarez et al., 2002) Suárez A. and Palomar M., *A maximum entropy-based word sense disambiguation system*, Proceedings, International conference on computational linguistics, 2002.

(Sussna, 1993) Sussna M., *Word sense disambiguation for free text indexing using a massive semantic network*, Proceedings, 2nd International conference on information and knowledge management, 1993.

(Tang et al., 2007) Tang F., Brennan S., Zhao Q., Tao H., *Co-training using semi-supervised SVM*, Proceedings ICCV 2007, pp: 1-8, 2007.

(Ting et al., 1997) Ting K. M. and Witten I. H., *Stacked generalizations: when does it work?*, Proceedings of the 15th IJCAI, Morgan Kaufmann, 1997.

(Vapnik, 1995) Vapnik, V. N., *The Nature of Statistical Learning Theory*, Springer-Verlag, 1995.

(Veronis, 2004) Veronis J., *Hyperlex: lexical cartography for information retrieval*, Computer Speech & Language, volume 18, issue 3, pp: 223-252, 2004.

(Volk, 2002) Volk M., *Using the web as a corpus for linguistic research*, publications of the Department of General Linguistics, University of Tartu, 2002.

(Wang et al, 2008) Wang P. and Domeniconi C., *Building semantic kernels for text classification using Wikipedia*, Proceedings, International Conference on Knowledge Discovery and Data Mining, pp: 713-721, 2008.

(Weiss et al., 1998) Weiss G.M. and Hirsh H., *Learning to predict rare events in event sequences*, Proceedings Fourth ICKDDM, pp: 359-363, 1998.

(Weiss et al., 1999) Weiss S. M., Apte C., Damerau F. J., Johnson D. E, Oles F. J., Goetz T. and Hampp T., *Maximizing text-mining, performance*, IEEE Intelligent Systems, volume 14, issue 4, pp: 63-69, 1999.

(Wolpert, 1992) Wolpert D. H., *Stacked generalization*, Neural Networks volume 5, pp: 241–259, 1992.

(Yavuz, 1974) Yavuz, D., *Zipf's law and entropy*, IEEE Transactions for Information Theory, volume 20, 1974.

(Yang et al., 1997) Yang Y. and Pedersen J., *A comparative study on feature selection in text categorization*, Proceedings, 14th International Conference on Machine Learning, pp: 412-420, 1997.

(Yang et al., 1999) Yang, Y., and X. Liu, *A re-examination of text categorization methods*, Proceedings, 22nd annual international ACM SIGIR Conference on Research and Development in Information Retrieval, pp: 42-49, 1999.

(Yarowsky, 1992) Yarowsky D., *Word sense disambiguation using statistical models of Roget´s categories trained on large corpora*, Proceedings, 14th international conference on computational linguistics, 1992.

(Yarowsky, 1995) Yarowsky D., *Unsupervised word sense disambiguation rivaling supervised methods*, Proceedings, 33rd Annual Meeting of the Association for Computational Linguistics, pp: 189-196, 1995.

(Zelikovitz et al., 2006) Zelikovitz S. and Kogan M., *Using web searches on important words to create background sets for LSI classification*, Proceedings, FLAIRS Conference, AAAI Press, pp: 598-603, 2006.

(Zeman et al., 2008) Zeman D. and Resnik P., *Cross-Language Parser Adaptation between Related Languages*, Proceedings, IJCNLP 2008 Workshop on NLP for Less Privileged Languages, 2008.

(Zhao et al., 2005) Zhao Y. and Zobel J., *Effective and Scalable Authorship Attribution Using Function Words*, Proceedings, 2nd Asian Information Retrieval Symposium, pp: 174-190, 2005.

(Zhu, 2005) Zhu X., *Semi-supervised learning literature survey*, Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005.