UN ALGORITMO SIMPLE Y EFICIENTE PARA LA CLASIFICACIÓN AUTOMÁTICA DE PÁGINAS WEB

María del Cisne García* **Fausto Pasmay** **Enrique V. Carrera**

Colegio de Ciencias e Ingeniería, USFQ

Resumen

Este artículo propone un simple pero eficiente clasificador de páginas Web basado en la frecuencia de términos. La simplicidad está dada por el uso de un conjunto pequeño de términos para describir cada clase, mientras que la eficiencia es alcanzada mediante embolsamiento. El uso de atributos simples como la frecuencia de términos también reduce la complejidad de los algoritmos de preprocesamiento y extracción de características. Sin embargo, un problema de usar propiedades dependientes de los términos incluidos en cada página es la selección de la descripción de términos correspondiente para cada una de las clases. En este trabajo, la selección de términos para cada clase se basa en el coeficiente TFIDF, mientras que el enbolsamiento utiliza clasificadores probados como redes neuronales y algoritmos bayesianos. Los resultados de nuestra evaluación muestran un clasificador sumamente rápido con una exactitud superior al 83 %.

Palabras Clave. Minería de datos, clasificación, frecuencia de términos, embolsamiento, *World Wide Web*.

Introducción

La clasificación o caracterización de páginas Web es el proceso mediante el cual se asigna una o más etiquetas predefinidas a cada documento expuesto en la Web. La tarea de clasificación es a menudo vista como un problema de aprendizaje supervisado en el cual un conjunto de datos previamente etiquetados es usado para entrenar un clasificador que puede posteriormente ser aplicado para etiquetar ejemplos futuros [1].

La clasificación de páginas Web es esencial para muchos procesos de administración y recuperación de información como el rastreo focalizado de páginas Web [2] y el desarrollo asistido de directorios [3]. La clasificación de páginas Web puede también mejorar la calidad de las búsquedas mediante el filtrado de contenido [4, 5] y la navegación asistida [6]. Basados en la importancia de estas aplicaciones y en el rápido crecimiento de la Web, consideramos que la clasificación automática de páginas asumirá un rol preponderante en los futuros servicios de búsqueda.

Sin embargo, la naturaleza incontrolable de los contenidos Web genera desafíos adicionales para la implementación de una clasificación correcta y eficiente [7]. Las páginas Web normalmente contienen ruido, como anuncios y barras de navegación, que impide la aplicación directa de la mayoría de métodos convencionales de clasificación. El ruido intrínseco de las páginas Web produce desviaciones significativas dentro de cualquier algoritmo de clasificación, haciendo que se pierda fácilmente la orientación sobre el tópico principal de su contenido.

Adicionalmente, el diseño de un clasificador requiere balancear el compromiso existente entre exactitud y eficiencia. Clasificadores sumamente exactos requieren algoritmos complejos y costosos, reduciendo su eficiencia desde el punto de vista de desempeño. Clasificadores rápidos, por otro lado, no son del todo exactos. De lo expuesto hasta el momento, además de perseguir exactitud en la clasificación, los clasificadores de páginas Web requieren considerar la complejidad de los algoritmos a implementar de manera que puedan mantener sus requisitos computacionales en niveles razonables. Esta condición es más crítica todavía si consideramos que la mayoría de aplicaciones mencionadas anteriormente requieren clasificar cientos o inclusive miles de páginas por segundo.

Fundamentados en esto, nosotros proponemos un clasificador de páginas Web que se basa en la frecuencia de términos, principalmente. El uso de un atributo tan simple como la frecuencia de términos permite reducir la complejidad del clasificador y alcanzar un buen desempeño. Además de alto desempeño, nuestro clasificador también busca exactitud y por ello emplea varios algoritmos ya probados como redes neuronales [8] y clasificadores bayesianos [9] combinados mediante la técnica denominada embolsamiento [10].

Con la finalidad de lograr los objetivos propuestos, tener una implementación modular y permitir una evaluación pormenorizada, nuestro clasificador de páginas Web consta de tres etapas claramente definidas: el preprocesamiento de las páginas, su clasificación misma y la etapa de entrenamiento de los clasificadores. La etapa de preprocesamiento consiste de una serie de filtros que extraen la mayoría de términos requeridos para discriminar las diferentes páginas, reduciendo la dimensionalidad de la información a tratar. Los algoritmos de preprocesamiento presentan una complejidad lineal con el

tamaño del documento. Por su parte, la tarea de clasificación aplica el conjunto de características sintetizadas en la fase de preprocesamiento a un conjunto de clasificadores previamente entrenados. Varios algoritmos de clasificación simples son agrupados mediante embolsamiento con la finalidad de aumentar la exactitud de la clasificación. Finalmente, la etapa de entrenamiento tiene por objetivo construir el modelo de clasificación usando ejemplos de páginas Web previamente categorizadas. Esta etapa es realizada una sola vez fuera de línea.

Antes de describir con mayores detalles la estructura propuesta para nuestro clasificador, revisemos algunos fundamentos teóricos usados en nuestro trabajo.

Fundamentos Teóricos

En esta sección se introducen los diferentes tipos de clasificación existentes en el contexto de las páginas Web, la forma de calcular el coeficiente TFIDF y la teoría detrás de la técnica de embolsamiento.

Clasificación de Páginas Web. El problema general de clasificar páginas Web puede ser dividido en múltiples subproblemas: clasificación temática, clasificación funcional, clasificación sentimental, entre otras [7]. Nuestro trabajo se centra en la clasificación temática, la misma que está orientada a distinguir el tópico principal de cada página Web.

Desde el punto de vista de la clasificación misma, esta tarea puede depender del número de clases existentes (clasificación binaria o de múltiples clases), del número de clases que pueden ser asignadas (clasificación con etiqueta única o de múltiples etiquetas), del tipo de asignación permitida (clasificación rígida o variable) y de la organización de las categorías (clasificación plana o jerárquica). El presente estudio se enfoca en la clasificación de múltiples clases usando una sola etiqueta cuya asignación es rígida (*i.e.,* no se permite estados intermediarios) y las clases presentan una estructura plana.

El Coeficiente TFIDF. El coeficiente TFIDF (*Term Frequency–Inverse Document Frequency*) [11] es una ponderación usada a menudo en tareas de recuperación de información y minería de texto. El coeficiente es una medida estadística usada para evaluar cuan importante es una palabra respecto a un documento perteneciente a una colección o cuerpo de documentos. La importancia de cada palabra incrementa proporcionalmente con el número de veces que ella aparece en el documento pero se ve influenciada por la frecuencia de la palabra en el cuerpo de documentos. Variaciones del esquema de ponderación TFIDF son usados a menudo por los motores de búsqueda como una herramienta para la puntuación y ranqueo de la relevancia de un documento ante una consulta de usuario determinada.

La frecuencia de un término (TF) en un documento dado es simplemente el número de veces que el término aparece en ese documento. Este valor es usualmente normalizado para prevenir que documentos extensos adquieran una inusual ventaja. De esta forma, la importancia del término t_i en el documento d_j está dada por:

$$\mathrm{TF}_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

donde $n_{i,j}$ es el número de ocurrencias de término considerado en el documento d_j, y el denominador es el número de ocurrencias de todos los términos en el documento d_j.

La frecuencia inversa de los documentos (IDF) es una medida de la importancia general del término y se calcula mediante:

$$\mathrm{IDF}_i = \log \frac{|D|}{|\{d_j : t_i \in d_j\}|}$$

donde el numerador es el número total de documentos en el cuerpo y el denominador es el número de documentos donde el término t_i aparece (*i.e.,* $n_{i,j} \neq 0$).

Así, el coeficiente TFIDF para el termino t_i en el documendo d_j es:

$$\mathrm{TFIDF}_{i,j} = \mathrm{TF}_{i,j} \times \mathrm{IDF}_i$$

Un valor TFIDF alto es alcanzado por un término con alta frecuencia en el documento considerado, pero baja frecuencia en la colección total de documentos. De esta manera, el coeficiente tiende a filtrar términos comunes.

Embolsamiento. Este es un concepto asociado al área de minería de datos y se aplica principalmente a la tarea de clasificación. La idea central del embolsamiento es combinar la salida de varios clasificadores o predictores individuales implementados muchas veces por técnicas de modelamiento distintas, pero entrenados bajo un mismo conjunto de datos. La combinación de la salida de los varios clasificadores se lleva a cabo mediante un simple mecanismo de votación.

Esta técnica se usa para contrarrestar las deficiencias de cada uno de los clasificadores individuales, además de reducir la inestabilidad inherente de los resultados cuando se tienen modelos complejos aplicados a conjuntos de datos pequeños. De esta forma, un clasificador basado en embolsamiento tiene normalmente una mayor exactitud que cualquier técnica de clasificación individual entrenada con el mismo conjunto de datos. Adicionalmente, el embolsamiento ayuda a incrementar la robustez del clasificador ante el ingreso de datos ruidosos, ya que el modelo compuesto reduce la varianza de los clasificadores individuales.

Breiman [10] inclusive mostró que el embolsamiento es principalmente efectivo en algoritmos de aprendizaje "inestables" como redes neuronales y árboles de decisión, donde pequeños cambios en el conjunto de entrenamiento producen grandes variaciones en su predicción.

Clasificador Basado en la Frecuencia de Términos

Nuestro clasificador incluye dos tareas fundamentales y completamente aisladas: el preprocesamiento de la página Web y su correspondiente clasificación. Sin embargo,

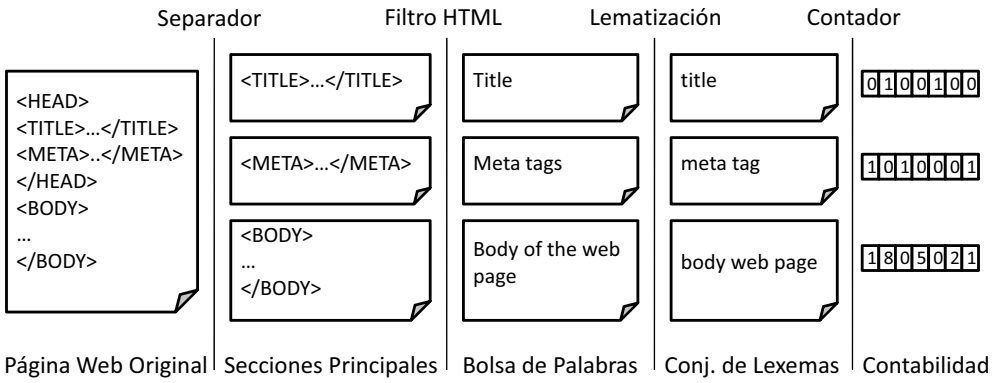

Figura 1. Preprocesamiento de una página Web

una tercera tarea llevada a cabo una sola vez es también incluida: el entrenamiento de los clasificadores. Estas tres tareas son descritas a continuación.

Preprocesamiento. La tarea de preprocesamiento puede ser descrita de una manera gráfica mediante el diagrama de la figura 1. Podemos ver que cada página Web es descargada y separada en tres secciones: (i) el título de la página, (ii) sus *meta* datos, y (iii) el cuerpo mismo del documento. Está división está justificada por la alta relevancia que tienen elementos como el título y los *meta* datos en la defición del tópico principal de una página Web [12]. Mediante esta división se permite ponderar dinámicamente la importancia relativa de cada sección.

Las tres secciones son entonces filtradas de manera independiente para eliminar todas sus etiquetas HTML. De esta forma, cada sección se convierte en una *bolsa de palabras* sin ningún tipo de formatación. Posteriomente, cada bolsa de palabras es procesada mediante el algorimo de Paice/Husk para encontrar los lexemas o raíces comunes de cada palabra. El uso de lexemas en lugar de sus variantes morfológicas tiene la ventaja de incrementar la tasa de asociación al agrupar términos con igual raíz a pesar de sus terminaciones diferentes.

Finalmente, los lexemas resultantes son filtrados conforme el conjunto de términos relevantes para la mayoría de categorías. El conjunto de términos relevantes para cada categoría es obtenido mediante la selección de los lexemas con mayor valor TFIDF dentro de su respectivo conjunto de páginas Web de entrenamiento. Cada categoría aporta con un número pequeño de términos y estos se encuentran agrupados en una lista única propia de cada sección. Usando la lista correspondiente de la sección procesada, este último filtro devuelve un vector con el número de veces que cada término relevante aparece en el conjunto de lexemas encontrados en la página Web. En otras palabras, cada uno de los componentes i del vector indica el número de veces que el término relevante t_i aparece en el conjunto de lexemas producidos.

Puede observarse que el conjunto de algoritmos usados por la tarea de preprocesamiento son procedimientos bien entendidos y relativamente baratos en términos de recursos computacionales. De hecho, la complejidad de la tarea de preprocesamiento es lineal con el tamaño

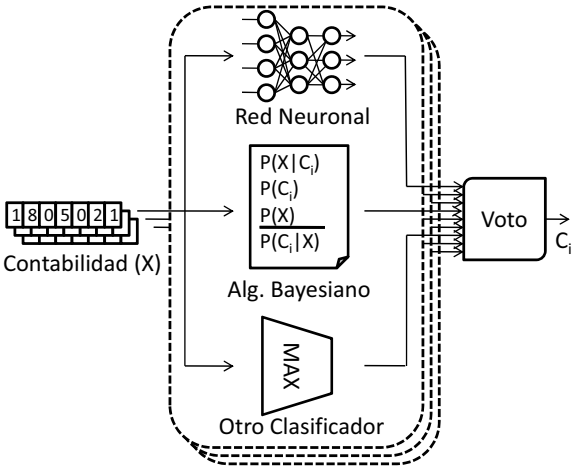

Figura 2. Clasificación mediante embolsamiento.

del documento HTML. Además, si bien las técnicas de preprocesamiento son simples, ellas son lo suficientemente inteligentes como para extraer la mayoría de características importantes de cada página. Es también importante notar que nuestra extracción de características reduce la dimensionalidad del problema al sintetizar un número pequeño de características relevantes. Esta reducción de dimensionalidad puede ser ajustada dinámicamente en función de la cantidad de términos que se escoja para representar cada clase.

Clasificación. La tarea de clasificación está esquematizada en la figura 2. Como se observa en dicha figura, cada vector contabilizando el número de lexemas relevantes dentro de cada página Web es presentado por separado a su conjunto de clasificadores (embolsamiento) de manera simultánea. Cada vector es entonces clasificado por tres modelos independientes previamente entrenados. En nuestro clasificador se han usado redes neuronales, algoritmos bayesianos y la mayoría de términos.

La red neuronal usa un esquema de recordación sin realimentación que fue entrenado mediante el algoritmo de retro-propagación del error. Retro-propagación es un método común de aprendizaje supervisado basado en el método del gradiente descendente [8]. Con relación a los algoritmos bayesianos, nuestro clasificador usa el clasificador ingenuo de Bayes, el mismo que aplica el teorema de Bayes asumiendo total independencia pro-

Figura 3. Interfase de clasificación

babilística entre sus variables. Este clasificador puede ser entrenado de manera bastante eficiente y trabaja mucho mejor de lo esperado en varias situaciones complejas del mundo real [9]. Finalmente, la mayoría de términos es un clasificador elemental que asigna una página a una clase usando únicamente el lexema con mayor repetición dentro de la contabilización generada por la tarea de preprocesamiento.

Puede observarse que la complejidad de clasificación de estas tres técnicas es bastante baja, especialmente el clasificador bayesiano y el uso de la mayoría de términos. En conjunto, a través del uso de embolsamiento se genera un clasificador bastante robusto y relativamente exacto que decide la clase final mediante una votación proporcional entre cada una de los clasificadores individuales y cada una de las secciones de la página Web.

Entrenamiento. Este es un proceso que se realiza fuera de línea por una sola ocasión. Es una tarea demorada y depende directamente del número de páginas Web de entrenamiento seleccionadas. De cualquier forma, el proceso de entrenamiento inicia obteniendo el conjunto de lexemas más relevantes a cada categoía. Para ello, cada página del conjunto de entrenamiento es preprocesada hasta convertirla en un conjunto de lexemas. Una vez que todas las páginas se han convertido en conjuntos de lexemas, se procede a calcular los coeficientes TFIDF de cada lexema existente en los docuementos de entrenamiento. Para finalizar esta primera parte, se ordenan los valores TFIDF dentro de cada categoría y se escojen los lexemas correspondientes a los valores TFIDF más altos.

El número de lexemas escogidos para cada categoría es un parámetro variable que sin duda generará resultados diferentes durante el proceso de clasificación de nuevas páginas Web.

Una vez escogidos los lexemas más relevantes a cada clase y combinados en listas únicas dependientes de la sección procesada, se terminan de preprocesar los conjuntos de lexemas usados para entrenamiento hasta convertirlos en vectores que contabilizan el número de lexemas relevantes que posee cada página. Estos vectores son entonces usados para entrenar tanto la red neuronal como el clasificador bayesiano, antes descritos.

Categorías	Aciertos (%)
Arte	88
Negocios	82
Computadores	86
Educación	86
Entretenimiento	84
Gobierno	80
Salud	84
Noticias	80
Deportes	82
Referencia	78
Ciencia	82
Sociología	80
Sociedad	84
Promedio	83

Cuadro 1. Resultados de exactitud.

Evaluación

Con la finalidad de evaluar el clasificador propuesto, hemos desarrollado un prototipo inicial basado en la plataforma Java. Más específicamente, se utilizó el JDK 1.6 ejecutando sobre una máquina Pentium 4 con 1MB de memoria RAM y con el sistema operativo Linux (kernel 2.6.24).

La implementación del sistema clasificador incluye seis clases fundamentales que permiten el uso de una interface gráfica como la presentada en la figura 3. El proceso de entrenamiento usa otras cuatro clases adicionales para el cálculo de los coeficientes TFIDF y el entrenamiento de los clasificadores correspondientes. Además de clasificar cualquier página Web individual, nuestro prototipo mantiene un directorio con todas las páginas ya clasificadas.

La configuración inicial del prototipo accesa un listado de 1300 documentos HTML previamente clasificados por el directorio de *Yahoo!* en 13 categorías básicas [3]. Cada categoría contribuye con 100 páginas Web al listado general. De todas las 1300 páginas pre-clasificadas, se usaron 910 (70 %) para el proceso de entrenamiento y las restantes 390 (30 %) para la verificación de exactitud del clasificador.

Durante el proceso de entrenamiento se escogieron únicamente los cinco términos más significativos a cada clase de acuerdo al valor de sus coeficientes TFIDF. De esta manera, se terminó generando vectores de 65 enteros para cada una de las 910 páginas de entrenamiento. Esos vectores fueron entonces usados para entrenar los diferentes clasificadores individuales que son usados mediante el esquema de embolsamiento.

El cuadro 1 muestra los resultados obtenidos con relación a la exactitud del clasificador una vez que las 390 páginas de prueba fueron aplicadas al sistema entrenado. Este cuadro resume los resultados porcentuales dentro de cada categoría y el valor promedio global.

Con relación al tiempo de ejecución, se pudo determinar que su mayor costo se encuentra en la descarga de

la página Web antes que en el procesamiento de la misma. En otras palabras, nuestra aplicación se encuentra al momento limitada por la latencia de la red antes que por la velocidad de procesamiento.

Conclusiones

El presente trabajo demuestra la importancia que está adquiriendo la tarea de clasificación de páginas Web y lo compleja que puede llegar a ser dicha tarea. Existe la necesidad de balancear exactitud con desempeño y hacia esa meta apunta el presente trabajo. Mediante la selección de técnicas de preprocesamiento simples se ha logrado extraer información crítica de cada uno de los documentos HTML y posteriormente discriminarlos con una exactitud bastante buena.

Como trabajos futuros en la misma dirección destacamos el refinamiento del prototipo inicial para permitir otros algoritmos de clasificación dentro del embolsamiento y la variación del número de características usadas para discriminar entre categorías diferentes.

Referencias bibliográficas

[1] Han, J. and Kamber, M. 2006. *Data Mining – Concepts and Techniques*, Morgan Kaufmann Publishers, San Francisco, CA, 2nd edition.

[2] Pant, G. and Menczer, F. 2003. Topical crawling for business intelligence. In ECDL pp. 233–244.

[3] Yahoo! 2008. Yahoo Directory. `http://dir.yahoo.com`.

[4] Ambrosini, L., Cirillo, V., and Micarelli, A. 1997. A hybrid architecture for user-adapted information filtering on the World Wide Web. In Proceedings of the 6th International Conference on User Modeling pp. 59–61.

[5] Paez, S., Pasmay, F., and Carrera, E. V. 2008. Improving personalized web search. Technical Report (*work in progress*). Department of Systems Engineering, University San Francisco of Quito.

[6] Joachims, T., Freitag, D., and Mitchell, T. M. 1997. Web Watcher: A tour guide for the World Wide Web. In IJCAI (1) pp. 770–777.

[7] Qi, X. and Davison, B. D. 2007. Web page classification: Features and algorithms. Technical Report LU-CSE-07-010. Department of Computer Science and Engineering, Lehigh University.

[8] Gupta, M. M., Jin, L., and Homma, N. 2003. *Static and Dynamic Neural Networks*, Wiley-Interscience, Hoboken, NJ, 1st edition.

[9] Zhang, H. 2004. The optimality of Naïve Bayes. In Valerie Barr and Zdravko Markov, (ed.), FLAIRS Conference, AAAI Press.

[10] Breiman, L. 1996. Bagging predictors. *Machine Learning*. 2(24), 123–140.

[11] Singhal, A., Salton, G., Mitra, M., and Buckley, C. 1996. Document length normalization. *Information Processing and Management*. 5(32), 619—-633.

[12] Shen, D., Chen, Z., Yang, Q., Zeng, H.-J., Zhang, B., Lu, Y., and Ma, W.-Y. 2004. Web-page classification through summarization. In Proceedings of the 27th Annual International Conference on Research and Development in Information RetrievalNew York, NY, USA: ACM Press. pp. 242–249.