Detección de Spam en la Web mediante el análisis de texto y de grafos

F. Javier Ortega, José A. Troyano, Fermín Cruz, and Fernando Enríquez

Departamento de Lenguajes y Sistemas Informáticos
Universidad de Sevilla
Av. Reina Mercedes s/n 41012, Sevilla (Spain)
{javierortega,troyano,fcruz,fenros}@us.es

Resumen El spam en la web representa un grave problema para los sistemas de Recuperación de Información, debido al perjuicio que puede ocasionar en la calidad de los resultados de los mismos. En este trabajo se presenta un sistema de detección de spam en la web basado en un algoritmo de ranking que ordena las páginas web de acuerdo a su relevancia, penalizando aquellas páginas susceptibles de ser consideradas spam. La novedad de este sistema reside en conjugar técnicas de procesamiento de textos con técnicas de análisis de grafos. Las técnicas de procesamiento de textos se utilizan para asignar a determinadas páginas una puntuación a priori, de acuerdo a la probabilidad de que sean spam o no, según su contenido. Nuestro algoritmo de ranking procesará el grafo de las páginas web y las puntuaciones a priori para obtener el ranking de webs. En los experimentos se comprueba que nuestro sistema mejora los resultados de otras técnicas muy utilizadas.

Keywords: Detección de Spam, Recuperación de Información, Algoritmos de Ranking

1. Introducción

El spam en la web consiste en la creación o modificación de páginas web con el objetivo de conseguir que los sistemas de recuperación de información indexen dichas webs, y obtener así un determinado beneficio gracias al aumento del tráfico web hacia esas páginas. Existen dos mecanismos de spam: el basado en contenido, y el basado en enlaces. El primero consiste en la modificación del contenido textual de las webs, generalmente añadiéndole a la misma una gran cantidad de palabras clave (*keyword stuffing*). El spam basado en enlaces trata de burlar a los sistemas de recuperación basados en el número de enlaces de las páginas webs. Una manera sencilla de hacer esto es crear un gran número de páginas web enlazadas entre sí, formando lo que se conoce como *granja de enlaces*. De esta forma, cada página creada tendrá un gran número de enlaces entrantes y, por tanto, su reputación se verá incrementada.

Existen muchos trabajos sobre detección de spam. En general se suelen enfocar hacia uno de los dos tipos de spam mencionados antes. Así podemos encontrar

sistemas basados en la detección de spam mediante el análisis de grafos. Dentro de este grupo se encuentra el TrustRank [5], basado en el cálculo de un ranking de páginas, utilizando un conjunto de páginas fiables seleccionadas a mano (semillas) para introducir un sesgo en la ejecución del algoritmo, de forma que dichas semillas tengan un mayor peso respecto al resto de páginas. Otro trabajo en esta línea es el presentado en [1].

Otras técnicas se basan en el contenido textual de las páginas para determinar si una web es o no spam. Estos métodos normalmente analizan la distribución de determinadas heurísticas sobre el contenido de páginas de spam y no spam, para generar un clasificador de documentos [4], [3]. Algunas métricas utilizadas para esta tarea son el número de palabras en una web, el contenido HTML invisible, la longitud media de las palabras de una página, etc.

Nuestro sistema combina conceptos de ambos tipos de técnicas, de forma que el contenido textual y los enlaces ayuden a detectar las webs de spam.

El resto del trabajo se organiza de la siguiente forma. En la Sección 2 explicamos nuestra propuesta. Seguidamente, en la Sección 3 se muestran los resultados experimentales de nuestro sistema. Finalmente en la Sección 4 presentamos las conclusiones y trabajos futuros.

2. Combinando enlaces y contenidos

Tanto las técnicas basadas en contenido como las basadas en enlaces han mostrado ser métodos fiables en la detección de spam, si bien ambos métodos presentan ciertas debilidades. Las técnicas basadas en contenido fallan a la hora de detectar granjas de enlaces. Por su parte las técnicas basadas en enlaces no tienen en cuenta el contenido de las páginas web en sus cálculos, lo que puede ocasionar que una página con un determinado contenido no deseado sea considerada relevante de acuerdo a sus enlaces.

Nuestro sistema consta de dos partes: un algoritmo de ranking y un conjunto de heurísticas basadas en el contenido. El objetivo de estas heurísticas es obtener cierta información a priori sobre la probabilidad de que una página sea o no spam, atendiendo a su contenido. Los valores de dichas métricas se incluirán dentro del algoritmo de ranking para añadir un sesgo en los cálculos, de forma que las páginas relacionadas con una web de spam vean disminuido su ranking global, y viceversa. En nuestro sistema hemos implementado dos métricas: la longitud media de las palabras de una página y el número de palabras repetidas. Por su parte, el algoritmo de ranking es una adaptación del PageRank [**?**], pero a diferencia de éste calcula dos puntuaciones para cada web: PR^+, que representa la relevancia de una página, y PR^-,que es la probabilidad de que dicha página sea spam. El ranking de webs se calcula según la diferencia entre PR^+ y PR^-. Estas puntuaciones se calculan siguiendo las Ecuaciones (1) y (2).

$$PR^+(v_i) = (1-d)e_i^+ + d \sum_{j \in In(v_i)} \frac{PR^+(v_j)}{|Out(v_j)|} \tag{1}$$

$$PR^{-}(v_i) = (1-d)e_i^{-} + d \sum_{j \in In(v_i)} \frac{PR^{-}(v_j)}{|Out(v_j)|} \quad (2)$$

donde v_i es un nodo del grafo (una página web), $In(v_j)$ es el conjunto de webs apuntando a v_j, y $Out(v_j)$ es el conjunto de nodos hacia los que apunta v_j. El algoritmo itera sobre los nodos del grafo, aplicando las ecuaciones (1) y (2), hasta que la diferencia máxima entre las puntuaciones de los nodos de dos iteraciones consecutivas sea menor que un umbral dado, t. Los vectores e_i^{+} y e_i^{-} son los encargados de incluir en el algoritmo las métricas basadas en contenido. De esta forma, según se inicialicen ambos vectores, podremos dar mayor importancia a algunas webs frente a otras. Estas webs son las *semillas* de nuestro algoritmo. Hemos implementado tres métodos de selección de semillas:

- Páginas Más Positivas y Páginas Más Negativas (MPN): seleccionamos como semillas las N páginas con mayores y menores valores de las métricas. Cada semilla se inicializa con un valor de $e_i = 1/N$.
- Más Positivas y Páginas Más Negativas con Métricas (MPN-M): se eligen como semillas las mismas webs que en el apartado anterior, pero sus puntuaciones se inicializan según los valores de las métricas, de forma que $e_i = Metricas_i/N$.
- Todas las Páginas como Semillas (TPS): se utilizan todos los nodos como semillas, aplicando la misma puntuación que la vista para MPN-M.

3. Experimentación y resultados

Dado que nuestro sistema no clasifica las páginas webs entre spam y no spam, sino que genera un ranking de las mismas, no podemos aplicar como método de evaluación las métricas típicas de los sistemas de clasificación. En su lugar, hemos implementado una técnica de evaluación muy utilizada en otros trabajos de detección de spam en la web, conocida como *PR-buckets* [5]. Este método da mayor relevancia a los fallos producidos por webs de spam que consiguen burlar al sistema colocándose en las primeras posiciones del ranking, dado que estas posiciones son las más importantes. Para tener una idea de la validez de los resultados de nuestro sistema, hemos ejecutado experimentos con una conocida técnica de detección de spam, TrustRank. Como hemos comentado, esta técnica se basa también en un algoritmo de ranking, aunque la selección de semillas se realiza a mano, al contrario que en nuestro sistema, en el que la selección de semillas es automática.

Los experimentos se han realizado con el corpus WEBSPAM-UK2006 Dataset [2], compilado expresamente para la tarea de la detección de spam en la web. El corpus está formado por unos 98 millones de páginas web, y unos 120 millones de enlaces. Un subconjunto de unos 11000 sitios web fueron etiquetados como spam o no spam, conteniendo un total de 10 millones de páginas de spam. Como marco de trabajo se ha utilizado el sistema de Recuperación de Información Terrier[1] para el indexado del corpus.

En la tabla 1 mostramos los resultados de nuestra técnica, con sus tres métodos de selección de semillas, junto con el algoritmo de TrustRank. Mostramos los resultados de los 10 primeros buckets, relativos a las primeras posiciones del ranking.

[1] `http://terrier.org/`

#	Páginas	TR	MPN	MPN-M	TPS
1	14	**0**	2	**0**	**0**
2	68	2	5	**1**	3
3	212	17	16	**4**	8
4	649	40	48	**21**	32
5	1719	73	104	**66**	101
6	3849	155	244	**124**	199
7	6513	254	392	**180**	297
8	9291	371	557	**255**	416
9	12102	448	742	**350**	537
10	14914	511	937	**440**	650

Cuadro 1. Número de errores acumulados para cada método: TrustRank (TR), Páginas Más Positivas/Negativas (MPN), Más Positivas/Negativas con Métricas (MPN-M) y Todas las Páginas como Semillas (TPS). El mejor resultado se muestra resaltado.

4. Conclusiones y trabajo futuro

En este trabajo hemos presentado un sistema de detección de spam en la web basado tanto en el contenido textual de las páginas como en los enlaces de las mismas. El sistema se ha evaluado con un corpus específico para la tarea de detección de spam, obteniendo muy buenos resultados, e incluso mejorando los de la técnica de TrustRank.

Como trabajos futuros nos planteamos la mejora de este método, implementando nuevas métricas basadas en contenido, y estudiando el impacto de dichas métricas en el rendimiento global del sistema, y en la complejidad en tiempo del mismo. También resultaría interesante evaluar la influencia del conjunto de semillas en los resultados finales del algoritmo, y estudiar posibles mejoras a los métodos de selección propuestos.

Referencias

1. Luca Becchetti, Carlos Castillo, Debora Donato, Ricardo Baeza-YATES, and Stefano Leonardi. Link analysis for web spam detection. *ACM Transactions on the Web*, 2(1):1–42, February 2008.
2. Carlos Castillo, Debora Donato, Luca Becchetti, Paolo Boldi, Stefano Leonardi, Massimo Santini, and Sebastiano Vigna. A reference collection for web spam. *SIGIR Forum*, 40(2):11–24, December 2006.
3. Gordon V Cormack, Mark Smucker, and Charles L A Clarke. Efficient and Effective Spam Filtering and Re-ranking for Large Web Datasets. *Computing Research Repository*, abs/1004.5, 2010.
4. Dennis Fetterly, Mark Manasse, and Marc Najork. Spam, damn spam, and statistics: using statistical analysis to locate spam web pages. In *Proceedings of the 7th International Workshop on the Web and Databases*, pages 1–6, New York, NY, USA, 2004. ACM.
5. Zoltán Gyöngyi, Hector Garcia-Molina, and Jan Pedersen. Combating web spam with trustrank. In *Proceedings of the Thirtieth international conference on Very large data bases*, volume 30, pages 576–587, Toronto, Canada, 2004. VLDB Endowment.