Advertising Keywords Recommendation for Short-Text Web Pages Using Wikipedia

WEINAN ZHANG and DINGQUAN WANG, Shanghai Jiao Tong University
GUI-RONG XUE, Aliyun.com
HONGYUAN ZHA, Georgia Institute of Technology

Advertising keywords recommendation is an indispensable component for online advertising with the keywords selected from the target Web pages used for contextual advertising or sponsored search. Several ranking-based algorithms have been proposed for recommending advertising keywords. However, for most of them performance is still lacking, especially when dealing with short-text target Web pages, that is, those containing insufficient textual information for ranking. In some cases, short-text Web pages may not even contain enough keywords for selection. A natural alternative is then to recommend relevant keywords not present in the target Web pages. In this article, we propose a novel algorithm for advertising keywords recommendation for short-text Web pages by leveraging the contents of Wikipedia, a user-contributed online encyclopedia. Wikipedia contains numerous entities with related entities on a topic linked to each other. Given a target Web page, we propose to use a content-biased PageRank on the Wikipedia graph to rank the related entities. Furthermore, in order to recommend high-quality advertising keywords, we also add an advertisement-biased factor into our model. With these two biases, advertising keywords that are both relevant to a target Web page and valuable for advertising are recommended. In our experiments, several state-of-the-art approaches for keyword recommendation are compared. The experimental results demonstrate that our proposed approach produces substantial improvement in the precision of the top 20 recommended keywords on short-text Web pages over existing approaches.

Categories and Subject Descriptors: H.3.3 [**Information Search and Retrieval**]: Search process

General Terms: Algorithms, Experimentation

Additional Key Words and Phrases: Contextual advertising, advertising keywords recommendation, topic-sensitive PageRank, Wikipedia

ACM Reference Format:
Zhang, W., Wang, D., Xue, G.-R. and Zha, H. 2012. Advertising keywords recommendation for short-text Web pages using Wikipedia. ACM Trans. Intell. Syst. Technol. 3, 2, Article 36 (February 2012), 25 pages.
DOI = 10.1145/2089094.2089112 http://doi.acm.org/10.1145/2089094.2089112

1. INTRODUCTION

In the last decade, online advertising has become a prominent economic force and the main income source for a variety of Web sites and services. According to Interactive

Part of the work is supported by NSF grants IIS-1049694, IIS-1116886, and a Yahoo! Faculty Research and Engagement Grant. G.-R. Xue is also supported by grants from NSFC project (no. 60873211) and RGC/NSFC project (no. 60910123).
Authors' addresses: W. Zhang (corresponding author) and D. Wang, Department of Computer Science and Engineering, Shanghai Jiao Tong University, no. 800, Dongchuan Road, Shanghai, China 200240; email: wnzhang@apex.sjtu.edu.cn; G.-R. Xue, Senior Director, Aliyun.com, Zhejiang, China; H. Zha, College of Computing, Georgia Institute of Technology, Atlanta, GA 30032.

DOI 10.1145/2089094.2089112 http://doi.acm.org/10.1145/2089094.2089112

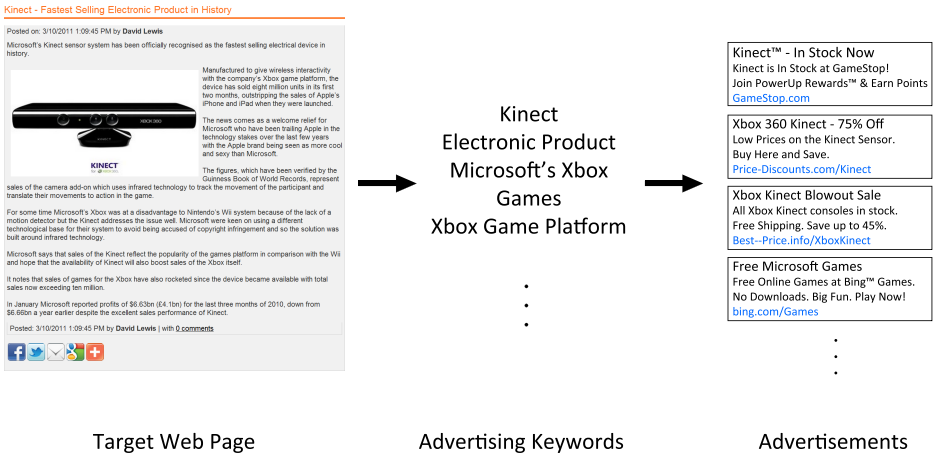

Fig. 1. An example of a keyword-based contextual advertising process.

Advertising Bureau's (IAB[1]) annual Internet advertising report, yearly advertising revenues have grown from US$6.01 billion in 2002 to US$26.04 billion in 2010 [IAB and PricewaterhouseCoopers 2011]. Some reports have already shown that online advertising has overtaken TV to become the largest advertising medium in countries such as the UK [Sweney 2009]. Among the several existing online advertising channels, search advertising, a search engine-based method of placing ads on Web pages, has become the driving force behind the large-scale monetization process of Web services through online marketing [Cristo et al. 2006].

Advertising keywords recommendation is an indispensable process of *sponsored search* and *contextual advertising*, which are the two main approaches of search advertising. In sponsored search, the ads are displayed at the top or the right of the result pages of search engines, largely based on the degree of matching between the keywords of user queries and an advertiser's bid keywords. The clicks on these ad links will take users to the Web pages of advertisers, generally called landing pages. A natural question for an advertiser is which keywords should be bid on for her landing pages? As users can express their search intent in a variety of different queries, it is almost impossible to conceive all the relevant keywords for the landing pages [Cristo et al. 2006]. Thus a sponsored search system, as an important service for the advertisers, provides recommendations of advertising keywords for the landing pages. In contextual advertising, on the other hand, it is also desirable to display relevant ads on the target Web pages [Anagnostopoulos et al. 2007]. This is mostly done by first extracting advertising keywords from the target Web pages and then retrieving the relevant ads using these advertising keywords. Figure 1 illustrates the keyword-based contextual advertising process. In summary, it is essential to accurately extract advertising keywords in order to display highly relevant ads, both in sponsored search and in contextual advertising.

Because of its importance, it is not surprising that a variety of approaches for advertising keywords recommendation for Web pages have been proposed including the supervised learning-based algorithm proposed by Yih et al. [2006], the KEA system [Fang et al. 2005; Jones and Paynter 2001; Witten et al. 1999], and the unsupervised learning algorithm proposed by Matsuo [2003]. However, these existing advertising keywords recommendation algorithms largely rely on the textual content of a Web

[1]Interactive Advertising Bureau. http://www.iab.net.

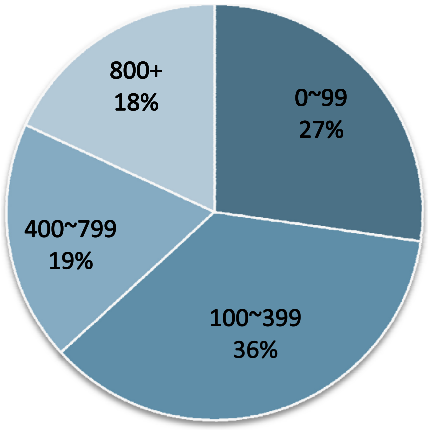

Fig. 2. Distribution of the word size of the Web pages in ODP.

page itself despite of the fact that a substantial number of Web pages mainly contain multimedia contents such as images or videos. As an illustration, we collected statistics about the distribution of word count of the Web pages from the Open Directory Project[2] in Figure 2. From Figure 2, we find that 27.24% pages in ODP have less than 100 words, which are generally called *short-text Web pages* in this article. If we consider ODP as a reasonable representation of an essential part of the whole Web, we can conclude that a substantial proportion of pages on the Web contain very little textual contents.

It is not surprising that traditional advertising keywords recommendation algorithms do not work well on these short-text Web pages. We can single out two main reasons: Firstly, the short-text Web pages offer less textual information. They probably contain very simple content structure and the content is poor, which makes it difficult for a recommendation system to rank the keywords well and thus leads to low accuracy. Secondly, in some cases, the situation is even worse; the short-text Web pages do not contain enough candidate terms or phrases.

A natural idea to overcome the preceding two issues is to enrich the set of advertising keywords by introducing new advertising keywords which do not occur in, but are still relevant to, the target Web pages. There are two possibilities. The first is to simply enrich the content of target Web pages and then use the enriched content to do the keywords recommendation work [Ribeiro-Neto et al. 2005]. The second one, which we develop in this article, requires to analyze the relationship between advertising keywords and then obtain new advertising keywords that are semantically relevant to the existing ones. So how can we identify the relationship between keywords? It turns out that Wikipedia[3] is an ideal resource to do that: It is a Web-based collaborative encyclopedia and contains, for example, more than 3 million entities in the English language. The entity articles cover a diverse set of topics in a large number of areas. And the number of its entities is still growing rapidly. Moreover, each entity is described by a relatively complete and concise article with hyperlinks linking to other Wikipedia entities indicating the semantic relationship between them. Therefore, they can be exploited to obtain high-quality advertising keywords relationships for recommendation.

In this article, we propose a novel approach of advertising keywords recommendation that makes use of entities and links from Wikipedia. As mentioned before, our

[2]Open Directory Project. http://www.dmoz.org.
[3]Wikipedia. http://www.wikipedia.org.

approach can recommend advertising keywords that are highly relevant to the target Web page even if they do not occur in it. These keywords are called *leveraged keywords* in this article. Our approach makes use of Wikipedia entites as the dictionary to recommend keywords. The usefulness of Wikipedia entities is evidenced by the fact that more than 99.8% of ODP Web pages contain one or more Wikipedia entities[4]. This high proportion makes Wikipedia a valuable thesaurus for Web pages.

Structurally, Wikipedia can also be viewed as a directed graph with vertices and edges corresponding to its entities and links among the entities, respectively. This allows us to generate the related advertising keywords by propagating the keywords on the graph using a Markov Random Walk. Specifically, we will use the algorithm of PageRank to implement the propagation process. Furthermore, inspired by topic-sensitive PageRank proposed in Haveliwala [2002], we introduce two kinds of bias, namely *content bias* and *advertisement bias*, into the propagation process, making it possible to recommend advertising keywords that are both relevant to a target Web page and valuable for advertising. In our experiments, we compare our algorithm to several baseline and state-of-the-art algorithms for advertising keywords recommendation. In particular, we focus on evaluating our algorithm on short-text Web pages. The result shows that our approach achieves substantial improvement over the supervised learning approach, measured by the precision of the top 20 recommended keywords, demonstrating the effectiveness of our algorithm.

The main contributions of our work are summarized as follows.

— The problem of the keywords recommendation for short-text Web pages is emphasized.
— A two-stage approach is proposed to solve the problem. In the first stage, candidate keywords are extracted from the target Web page while in the second stage, related keywords to the target Web page are recommended, using a random walk-based algorithm applied to the Wikipedia graph. The experimental result shows a significant improvement on the recommendation performance for short-text Web pages.
— To the best of our knowledge, our proposed content- and advertisement-sensitive PageRank is the first of its kind for multitopic-sensitive PageRank algorithms.

The rest of this article is organized as follows. In Section 2, we discuss several related works about search advertising, keywords recommendation, and application of Wikipedia. Some preliminary works are presented in Section 3. In Section 4, we present our approach using the content- and advertisement-sensitive PageRank on the Wikipedia graph. In Section 5, we describe the experiments and analyze the results. Finally, we present our conclusion and future work in Section 6.

2. RELATED WORK

2.1. Search Advertising

According to the Internet Advertising Revenue Report for the year 2010 [IAB and PricewaterhouseCoopers 2011], 45% of the total revenue from online advertising in the United States is contributed by search advertising, which continues to lead in the market and is followed by Display Banners (26%) and Classifieds (9%). Searching advertising is a search engine-based approach of placing online ads on Web pages. It is an interesting subfield of Information Retrieval that involves large-scale search, content

[4]This statistic work is made by us.

analysis, information extraction, statistical models, machine learning, and microeconomics. Described by IAB, the two main forms of search advertising are *sponsored search* and *contextual advertising*.

2.1.1. Sponsored Search. When a query is submitted to the search engine, two searches are performed. The first one is *organic search* which returns the Web pages with relevant content. The second one is sponsored search which returns the paid ads [Becker et al. 2009]. Sponsored search was introduced by Overture[5] in 1998 and now Google[6] offers the largest service. The search engine retrieves the ads of sponsors mainly by the keywords of the user query and displays them on the top or right of the search result pages. Generally, there are three forms of cost: *Cost-Per-Click* (CPC), *Cost-Per-Mille* (CPM), and *Cost-Per-Action* (CPA). The sponsored search system makes auctions on every keyword and the advertisers bid on some keywords for their ads. It is more likely for their ads to be ranked higher if the advertisers pay more for the impressions and clicks of their ads.

The ranking mechanism for sponsored search decides which ads retrieved should be ranked higher. In the work of Feng et al. [2003], two mainstream ranking mechanisms are compared: *ranking by Willingness To Pay* (WTP) and *ranking by Willingness To Pay* $\times$ *Relevance* (WTP $\times$ Rel). Through computational simulations, they found WTP $\times$ Rel performs better in almost all cases, while WTP is better when the correlation between the relevance and WTP is large.

Besides the works on the ranking mechanism, more academic research focuses on the matching strategy for the improvement of the relevance between ads and user queries [Hillard et al. 2010; Broder et al. 2008; Raghavan and Hillard 2009]. Since the content of ads and user queries are both short, short content matching algorithms are used. Some query expansion-based work is presented in Section 2.2.2. Other work makes use of some external information such as the content and types of landing pages [Becker et al. 2009; Choi et al. 2010].

2.1.2. Contextual Advertising. Contextual advertising, introduced by Google[7] in 2002, refers to the placement of ads on third-party Web pages based on the content of the target Web pages and the ads. The publishers and search engine will share some revenue once any ad on their Web pages is clicked. Some studies [Wang et al. 2002] have already shown that the relevance between the content of target Web pages and the ads makes a large difference in the click-through rate. Intuitively, the content of target Web pages suggests the users' interest and if the ads are relevant to the Web page content, they are more likely to attract users. Therefore, the matching work of the target Web pages and the ads is the key point of contextual advertising.

Keyword-based approaches are widely used in contextual advertising. This kind of approach first extracts keywords from the target Web pages and then uses these keywords to retrieve the ads just like sponsored search. However, due to the vagary of keywords extraction and the lack of Web page content, keyword-based approaches always lead to irrelevant ads. The work of Ribeiro-Neto et al. [2005] is a typical keyword-based approach, which matches the ads with the target Web pages' content and extracted keywords to get the winning strategy. Besides keyword-based approaches, the authors in Broder et al. [2007] make use of semantic information to enhance the matching work. They classify both pages and ads into a common taxonomy and merge the keywords matching work with the taxonomy matching work together to rank the ads.

[5]Overture. http://www.overture.com.
[6]Google Adwords. http://adwords.google.com.
[7]Google AdSense. http://www.google.com/adsense.

As analyzing the entire page content is costly and thus new or dynamically created Web pages could not be processed to match the ads ahead of time, the authors in Anagnostopoulos et al. [2007] proposed a summarization-based approach to enhance the efficiency of contextual advertising with an ignorable decrease in effectiveness.

2.2. Keywords Recommendation

Generally speaking, keywords recommendation refers to finding the relevant keywords to a given target for some application. According to the type of the target, two major keywords recommendation problems are Web page keywords recommendation (also called keywords extraction) and related keywords recommendation.

2.2.1. Web Page Keywords Recommendation. As an important part of contextual advertising, Web page keywords recommendation is indispensable. In this process, valuable advertising keywords are automatically found to match the ads. It is obvious that the more accurate and valuable these found keywords are, the more relevant the ads will be delivered on the Web pages. Two kinds of approaches for Web page keywords recommendation are widely used currently. One is supervised learning and the other is unsupervised learning.

In supervised learning, a set of example pages that have been labeled with keywords by human editors are given as the training data. Features of each word should be carefully selected. There are several approaches, such as the traditional TF $\times$ IDF model, GenEx system [Turney 2000], the KEA system [Fang et al. 2005], and Yih's et al.'s approach [2006]. GenEx system is one of the best known programs for keywords recommendation. It is a rule-based approach with 12 tuned parameters and is well used for pure textual content such as journal articles and email messages. However, keywords recommendation for Web pages should be considered more. Web pages contain various content structure with all kinds of multimedia information. Features of the Web page should be taken into consideration to train the supervised learning model. The improved KEA [Turney 2003] and Yih et al.'s [2006] approach bring various Web-related features such as the metadata, URL, anchor, and so on. In the experiment, we select Yih et al.'s [2006] approach as the baseline keywords recommendation approach and more details will be presented in Section 4.3.1. In the work of Ravi et al. [2010], candidate bid phrases are generated through a translation model and then well-formed ones get ranked up by a language model.

The general idea of unsupervised learning is to create some appropriate formulas based on the features similar to supervised learning to score these candidate keywords. Those who have scores in top k are recommended as keywords. In the work of Litvak and Last [2008], the authors proposed to use the HITS algorithm [Kleinberg 1999] to get the importance of the blocks (words, phrases, sentences, etc.) in lexical or semantic graphs extracted from text documents. On the other hand, Matsuo proposes a new approach of unsupervised learning based on the co-occurrence information to optimize the recommendation result [Matsuo 2003]. Moreover, some approaches trying to enrich the target Web page content to improve the performance are also proposed, such as the work of Ribeiro-Neto et al. [2005].

Although these approaches are effective in the experiments and have been widely used, a problem remains that these approaches highly depend on rich structure or content of the target Web pages. Thus for the short-text Web pages, these approaches can hardly provide high performance.

2.2.2. Related Keywords Recommendation. Given a target keyword, related keywords recommendation refers to finding the relevant keywords, such as synonyms,

semantically relevant phrases, and some rewrite from the target keyword. It has been widely used in the field of information retrieval and search advertising.

In the information retrieval field, query expansion or query substitution is an important topic. The raw user queries will be processed to be substituted by one or a list of keywords to obtain better search results [Jones et al. 2006; Mitra et al. 1998]. Much has been done about query substitution. Boldi et al. [2008, 2009] make use of the user search session data to build a query flow graph and then use the random walk on the graph to get related queries. Besides the session data, the search engine click-through data is used for mining the similarity between queries in the work of Cao et al. [2008].

There are also many works about related keywords recommendation for a given keyword in search advertising. In search advertising, broad match helps indirectly match the user queries with the advertising bid keywords. Most of state-of-the-art matching algorithms expand the user query using a variety of external resources, such as Web search results [Abhishek and Hosanagar 2007; Joshi and Motwani 2006; Radlinski et al. 2008], page and ad click-through data [Antonellis et al. 2008], search sessions, taxonomy [Broder et al. 2009] or concept hierarchy [Chen et al. 2008]. In addition, Radlinski et al. [2008] consider more about the feasibility of matching ads and the search engine revenue. Wang et al. [2009a] improve the efficiency of query expansion for sponsored search by proposing a novel index structure and adapting a spreading activation mechanism.

2.3. Application of Wikipedia

Wikipedia is a free online encyclopedia in which large set of concepts are well expressed by experts and volunteers. It provides a considerable knowledge base, covering areas such as art, history, society, and science. Wikipedia is considered as an ideal knowledge base for not only readers and researchers to look up knowledge but also for modern data mining systems to find auxiliary data to improve performance. Specifically, the articles of each Wikipedia entity contain a detailed explanation from various aspects. Moreover, the content of these articles is organized in well-structured format. This advantage can help automatic learning systems easier fetch information of entities. Furthermore, lots of links in the corpus of entities can imply a semantic relationship between linked entities, which can help automatic concept recognizers find related information.

Because of the diversity of content and the structured information [Medelyan et al. 2009], Wikipedia has attracted more and more researchers taking these advantages on the typical topics. Besides the application on keywords recommendation as introduced in this article, many improvements have been achieved in other areas. Schönhofen [2006] exploits the titles and categories of Wikipedia articles to identify the topics of documents. In the work of Carmel et al. [2009], Wikipedia is applied to enhance cluster labeling and the authors claim that using Wikipedia entities to label each cluster outperforms using keywords directly in the text. Wang et al. improve text classification by enriching the document with the entities of Wikipedia [Wang and Domeniconi 2008; Wang et al. 2009b]. Hu et al. map the target to a Wikipedia thesaurus and use the entity content and links to enhance the query intent identification [Hu et al. 2009] and text clustering [Hu et al. 2008]. Yu et al. evaluate ontology based on categories in Wikipedia [Yu et al. 2007].

3. PRELIMINARIES

Before we discuss our algorithm, we first introduce some basic materials to set the stage for further discussion.

3.1. Topic-Sensitive PageRank

The PageRank algorithm [Brin and Page 1997; Brin et al. 1998; Page 1997] is a widely used algorithm to obtain the *static* quality of Web pages based on the link graph. The basic idea is to perform a random walk on the link graph and propagate the quality score of a Web page to the ones it is linked to. Formally, the PageRank can be characterized by the following iteration

$$\vec{R}_{m+1} = (1-d)\vec{B} + dG \cdot \vec{R}_m, \tag{1}$$

where $\vec{R}_m = [r_1^{(m)}, \cdots, r_n^{(m)}]^T$ is a vector of PageRank of the whole indexed Web pages on the mth iteration and $r_i^{(m)}$ stands for the PageRank score of the Web page i. Moreover, $\vec{B} = [\frac{1}{n}]_{n\times 1}$ is the damping vector and the decay factor α limits the effect of the propagation of PageRank. The matrix G represents the row-normalized adjacency matrix of the link graph, that is, if there is an edge from vertex i to vertex j, then $G_{j,i} = \frac{1}{o_i}$, where o_i represents the out-degree of vertex i.

The topic-sensitive PageRank proposed in Haveliwala [2002] extends PageRank by allowing the iteration process to be biased to a specific topic. Specifically, the damping vector $\vec{B}$ is biased to the specific topic so that the Web pages of this particular topic are more likely to have high scores. The iterative scheme is the same as Eq. (1) except for the damping vector $\vec{B} = [b_1, \cdots, b_n]^T$ with the damping value b_i indicating the relevance of page i to the topic in question. It is easy to see that the propagation process is biased by the damping vector $\vec{B}$ at each iteration. Consequently, pages with higher damping values can propagate higher scores to their neighbors in the link graph.

3.2. Wikipedia Graph

In this section, we consider the construction of the Wikipedia graph. Firstly, we take each entity as a vertex in the graph. Then we aim at connecting two entities with an edge of the graph if they are semantically related. To this end, we notice that there are many hyperlinks in each Wikipedia article linking to the pages of other articles of Wikipedia entities. These hyperlinks are just potential edges for the graph we want to construct, because many Wikipedia articles link to (other articles about) dates and regions or other entities that are general but otherwise semantically unrelated. To address this issue, we make use of the entity category information. Specifically, we remove the edges between the entities of different first-level categories in the Wikipedia category tree. This way, there are only edges linking the entities in the same topic. Therefore, the Wikipedia graph reduces to κ subgraphs, where κ stands for the number of first-level categories, specifically $\kappa = 10$ for a Wikipedia version in January 2010. Additionally, we also consider weighting the edges by the number of links between two entities. This is reasonable because more edges from entity i to entity j means entity j is more related than other neighbors of entity i. Therefore, given the whole set of Wikipedia articles, we can construct a directed graph, which is called *Wikipedia Graph*. In the rest of the article, we will denote the row-normalized Wikipedia graph link matrix as $\mathcal{G}$, which is an $n \times n$ matrix where n stands for the size of the entity set. Element $\mathcal{G}_{i,j}$ of the matrix is given by

$$\mathcal{G}_{i,j} = \frac{\eta(j,i)}{\sum_{k=1}^{n} \eta(j,k)}, \tag{2}$$

where $\eta(j,i)$ denotes the edge number from entity j to entity i. Figure 3 gives an illustration of a Wikipedia subgraph.

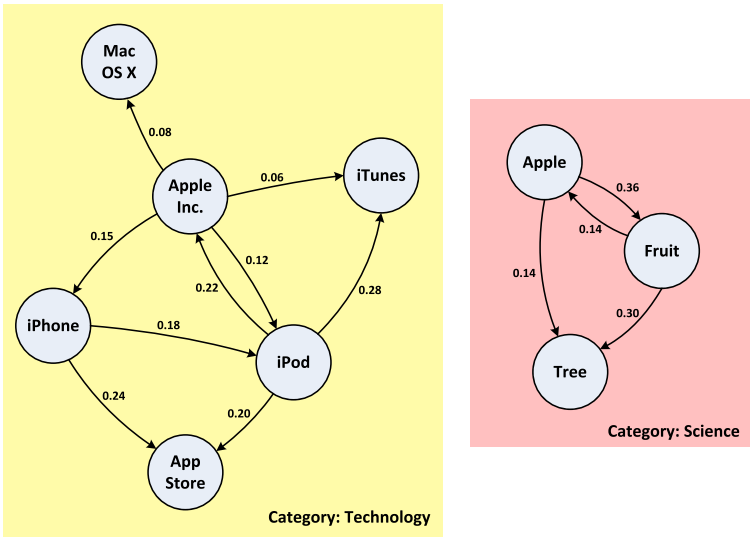

Fig. 3. An illustration of a Wikipedia subgraph.

4. CONTENT- AND ADVERTISEMENT-SENSITIVE PAGERANK

4.1. Problem Definition

The problem we need to address is: Given a target Web page p as the input, particularly, one with short-text content. The output should be k ranked keywords for p, required to be both relevant to the content of p and valuable for advertising.

4.2. Algorithmic Details

Here we describe the details of our algorithm of content- and advertisement-sensitive PageRank on the Wikipedia graph.

As our problem focuses on short-text Web page, the textual information of these pages themselves tends to be very limited. This information, however, is still an indispensable part for the proposed algorithm. Specifically, the textual content of p is first parsed and the Wikipedia entities occurring in p are identified with the same approach in Wang and Domeniconi [2008]. Then those entities are scored using traditional approaches such as, in our experiment, the supervised learning approach similar to Yih's work [Yih et al. 2006]. The score, defined as *content relevant score*, indicates the relevance of the entity to the content of p and measures the possibility of the entity to be a keyword of p.

Secondly, the output keywords of our algorithm should be valuable for advertising. Intuitively, the frequency of an entity name in some advertisement content can be a criterion to measure its advertising value. Therefore, another score on each entity is given, which we term as *advertisement relevant score*, defined as the percentage of occurrences of each entity name in a given set of ads. This score measures how likely the entity will occur in advertisement texts and can be used to retrieve the corresponding advertisement. In addition, this score is not based on the input target Web page, and thus can be calculated offline. As a consequence, every entity has two topic scores: the content relevant score and the advertisement relevant score.

Thirdly, we use the two scores in the content- and advertisement-sensitive PageRank iterations. In this process, the entities that are related to the content of target Web pages and valuable for advertising will get higher score because of the two

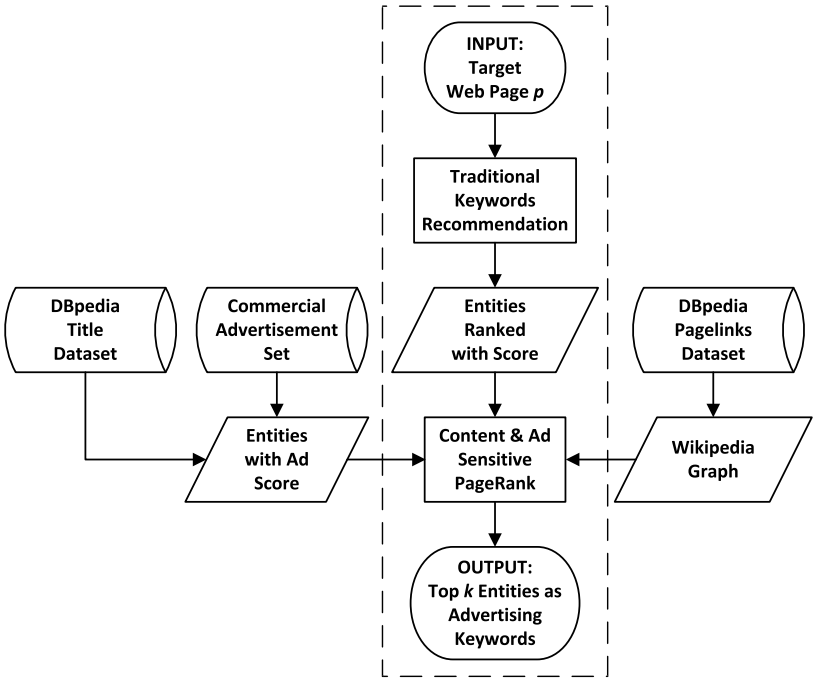

Fig. 4. Algorithm flow chart. The online process is in the dotted box and the offline one is outside.

scores on content and advertisement bias. The neighbors of these entities will also get a higher score because of the propagation process of the PageRank score.

Finally, we rank and choose k entities with the highest content- and advertisement-sensitive PageRank score as the output of our algorithm. The overall steps of the algorithm are depicted in Figure 4.

We next describe the details of the content relevant score and the advertisement relevant score in Section 4.3. Utilizing the two scores as the damping vectors, the work of content and advertisement PageRank is described in Section 4.5.

4.3. Damping Vector Setup

Considering that the recommended keywords should be both relevant to the target Web pages and valuable for advertising, we define two factors for each entity, corresponding to two types of damping vectors in the expression of topic-sensitive PageRank.

4.3.1. Content Damping Vector Setup. Given the target Web page p, for each entity i, its content damping value c_i stands for the relevance between entity i and the content of p.

We used a supervised learning approach to score each entity and determine whether an entity has a score high enough to be treated as a keyword. In our approach, we implement the approach of Yih et al. [2006] to generate the feature vector for each entity in p. Firstly, p is parsed and Wikipedia entities in p are extracted. Then these entities in p are scored by a regression approach. The regression approach is implemented by a *Support Vector Machine*, which has been trained with a large set of Web pages with keywords extracted by human editors. The entity with a higher score implies that it is more important to p. The features we selected for regression are listed in Table I.

Table I. Features Selected for Supervised Learning

Name	Type	Description
URL	Boolean	Whether it is in the page URL
Title	Boolean	Whether it is exactly the title
Headline	Boolean	Whether it is exactly the headline
Anchor	Boolean	Whether it is a the anchor text
TF	Double	The frequency of the entity in the page (normalized)
Link	Boolean	Whether it is part of a hyperlink of the page
PartLink	Boolean	Whether part of the entity is part of a hyperlink of the page
Meta	Boolean	Whether it is in the meta text
Span	Boolean	Whether it is in the span text
OneCapt	Boolean	Whether the first word of the entity is capitalized
AllCapt	Boolean	Whether each word of the entity is capitalized
Length	Double	The string length of the entity (normalized)
QueryLog	Boolean	Whether the entity is in the query log from AOL

The last feature in the list, QueryLog, which is proposed in Yih et al.'s work [2006], appears to be a novel feature to help improve the performance of supervised learning. It is claimed that the entities that occur in the user query are more likely to be keywords since the users use them in the search engine to retrieve some pages they want. In our experiment, we use the query log from AOL[8].

These features are of different importance, for example, an entity appearing in the title is more likely to be a keyword than an entity just appearing somewhere in the content body. Therefore, we weight those features and tune the weight to obtain the best performance.

4.3.2. Advertising Damping Vector Setup. Using advertisement-sensitive PageRank, we can obtain more commercial entities which are suitable for advertising keywords. The entities which are more relevant to the advertisement topic should have higher advertisement damping value. In our approach, we record the frequency of each Wikipedia entity i in the text of an advertisement set, defined as a_i. For the calculation of the PageRank iteration, we let a_i be the damping value biased to the advertisement for each entity i.

4.4. Two Topic-Sensitive PageRanks

Based on the topic-sensitive PageRank and the two topic relevant scores, we can construct two topic-sensitive PageRanks: *content-sensitive PageRank* and *advertisement-sensitive PageRank*. Given the content relevant score c_i for each entity i to the target Web page p, we can obtain the *content damping vector* $\vec{C}$. The iteration formula of content-sensitive PageRank is

$$\vec{R}_{m+1} = \alpha\vec{C} + (1-\alpha)\mathcal{G} \cdot \vec{R}_m, \tag{3}$$

where $\vec{R}_m = [r_1^{(m)}, \cdots, r_n^{(m)}]^T$ is the vector of the PageRank scores of all the entities on mth iteration and the parameter α controls the impact of the content relevant score to the content-sensitive PageRank value[9]. Similar to topic-sensitive PageRank on Web pages, this process can propagate the relevance scores from the seed entities with high

[8]AOL search data mirrors. http://www.gregsadetsky.com/aol-data/.

[9]Here α corresponds to $1-d$ in Eq. (1). Using this notation makes it seamless to merge the two topic-sensitive PageRanks into a multitopic-sensitive one, as will be discussed later.

content relevant scores to other relevant entities even if they do not occur in the target Web page. As a result, we can enrich the keywords set of the target Web page with the help of content-sensitive PageRank.

Similarly, given the advertisement relevant score a_i for each entity i, we can obtain the *advertisement damping vector* $\vec{A}$. The iteration formula of advertisement-sensitive PageRank is

$$\vec{R}_{m+1} = \beta\vec{A} + (1-\beta)\mathcal{G} \cdot \vec{R}_m, \tag{4}$$

where β is the parameter that controls the impact of the advertisement relevant score to the advertisement-sensitive PageRank value. Likewise, the entities that are related to the advertisement topic are more likely to get higher scores after every iteration by using the impact of advertisement damping vector $\vec{A}$. In addition, neighboring entities can share those scores which may also be relevant to the advertisement. Thus, the advertisement bias is incorporated into the random walk process.

4.5. Content- and Advertisement-Sensitive PageRank Algorithm

As we emphasized before, the recommended keywords should be both relevant to the target Web page and valuable for advertising. These two requirements can be satisfied by the two topic-sensitive PageRanks introduced earlier. We propose an approach that combines the two different topic-sensitive PageRanks to simultaneously address the preceding two requirements. The iteration formula combining the two topic-sensitive PageRanks is

$$\vec{R}_{m+1} = \alpha\vec{C} + \beta\vec{A} + (1-\alpha-\beta)\mathcal{G} \cdot \vec{R}_m. \tag{5}$$

In the previous equation, there are two damping vectors: $\vec{C}$, biased to the target Web page content, and $\vec{A}$, biased to the advertisement topic. Intuitively, in each iteration, the PageRank of an entity that is both relevant to the target Web page content and the advertisement topic will get a relatively high score boost by the factor $\alpha c_i + \beta a_i$, and each entity will also distribute its score to its neighbors. Therefore, entities that are relevant to both the content of the target Web page and the advertisement topic can obtain higher scores after the convergence of the iteration process.

4.6. Computational Details

4.6.1. Initial PageRank Vector Setup. As is well-known the convergence of PageRank will not depend on the initial value for each entity (the element of the start vector $\vec{R}_0$). The number of iterations and thus execution time, however, are greatly affected by the initial values. Thus an appropriately chosen initial value for each entity will improve the efficiency of the PageRank iteration process.

We propose to set the initial vector as

$$\vec{R}_0 = \alpha\vec{C} + \beta\vec{A}, \tag{6}$$

where α and β are the same parameters in Eq. (5) while $\vec{C}$ and $\vec{A}$ have been determined in Section 4.3.1 and Section 4.3.2. In our experiment, it takes about 25% less time to get convergence, compared with setting the initial vector with the uniform value.

4.6.2. Computational Complexity. In this section, we discuss the computational complexity of content- and advertisement-sensitive PageRank. The computation process $(1-\alpha-\beta)\mathcal{G} \cdot \vec{R}_m$ theoretically takes an $O(n^2)$ time. Since we have removed the cross-category edges when constructing the Wikipedia graph, we need not traverse all the entities in each iteration process; instead we just traverse the subgraphs in the categories of the seed entities. Therefore, in each iteration, $O(\overline{n})$ vertices are involved,

where $\overline{n} = n/\kappa$ stands for the average number of entities in each subgraph, and κ has been mentioned in Section 3.2. Furthermore, the Wikipedia graph is a sparse graph with each entity having on average 18.3 in-edges, thus $\mathcal{G}$ is a sparse matrix where each row has on average no more than 18.3 nonzero numbers. Taking advantage of this, we only record the in-edge neighbors for each entity and thus the calculation process reduces to an $O(\overline{n} + \overline{n} + 18.3\overline{n})$ time.

Given the initial PageRank vector, the number of iterations is still dependent on the two parameters α and β. Generally, the larger $(1 - \alpha - \beta)$ is, the more slowly the convergence is. In our experiments, six iterations to reduce the fluctuation of PageRank value $\triangle r_i^{(m)} = |r_i^{(m+1)} - r_i^{(m)}|$ under 10^{-4} when setting $\alpha = 0.85$ and $\beta = 1.5 \times 10^{-5}$. In our experiment, the average real runtime for each test case is 8.26 seconds (Java Platform, Windows, 2GB memory, 2.6 GHz). Furthermore, the efficiency can be improved with the optimization work discussed in the next subsection.

4.6.3. Optimization for Efficiency Improvement. Several optimizations can be incorporated for accelerating the computation. In this subsection, we discuss the optimization work which has been implemented in our experiment.

In PageRank calculation, it is time consuming for updating the PageRank value of the whole entity set. For our model, the number of entities with nonzero initial value is extremely low (20, in our experiment). Thus in each calculation, those entities with zero value and without nonzero neighbors can be ignored. We just consider the involved entity set, in which the entity PageRank value will be changed. Before the ith iteration of the computation, define the set of entities with nonzero PageRank score as S_{i-1}. It is clear that in this step of computation, only the entities in S_{i-1} or the ones with an in-edge from S_{i-1} will have a nonzero entry. Therefore, in the ith iteration of the computation, we only consider S_i in the computation instead of all the entities in the WikiGraph. In our experiment, the average real runtime for each test case is reduced to 2.95 seconds with this optimization.

5. EXPERIMENTS

This section mainly discusses the experimental results for our proposed algorithm, including the data preparation, comparisons with existing algorithms, evaluation metrics, performance results, and further discussions.

5.1. Data Preparation

5.1.1. Wikipedia Graph. *DBpedia*[10], according to its description, is a community effort to extract structured information from Wikipedia and to make this information available to the general public. The structured information can be downloaded from its Web site.

In our experiment, we take the *Pagelinks* dataset with the date of September 2009. It has 81.83 million triples and each triple represents a relation where the first entity has a page link to the second entity, resulting an initial graph with 81.83 million directed edges and 9.54 million entities. We refine the graph by removing the entities without out-edges and the ones which have some characters with ASCII $>$ 127. As mentioned in Section 3.2, we also removed the cross-category edges with the help of the *Article Categories* and *Categories(Skos)* dataset on DBpedia. With this preprocessing, the entity number reduces to 3.12 million and the edge number 57.29 million, as is shown in Table II.

[10]DBpedia. http://dbpedia.org/.

Table II. Data of Wikipedia Graph

Release Date	Vertice(entities)	Edges(pagelinks)	Average Out-degree
Sept. 2009	3.12 million	57.29 million	18.3

5.1.2. Advertisement Set. We use a set of 9 million textual ads, which are crawled by a commercial search engine using the AOL query log data. Each advertisement consists of the title and the description of the ads, which contain up to 10 and 30 words, respectively. Both of the two sections of the ads are composed by advertisers.

5.1.3. Training Data. We use 6422 human-labeled Web pages as our training pages. We extract the Wikipedia entities from the training pages and construct the vector for each entity with the feature described in Table I. Since those entities have been labeled, we can use these vectors with labels to train a classifier.

5.1.4. Test Data. We use the corpus of ODP Web pages as test pages in our experiments. The number of sampled test pages should be determined so that the experiments can convincingly demonstrate the difference of performance among the algorithms compared. In addition, the test pages should cover more topics so as to be representative. In our experiments, we generated two test datasets. The first dataset consists of 100 randomly selected short-text[11] ODP Web pages, called the *short-text Web page set*. The second one consists 103 Web pages, with 50 in short text and the other 53 in moderate or long text. We call the second test dataset the *overall Web page set*. These Web pages cover the topics of business, agriculture, art, computers, entertainment, automobiles, sports, Internet, life products, medicine, music stars, and so on.

5.2. Algorithms Used for Comparison

To demonstrate the effectiveness of our approach, we use some other approaches as the baseline or control in our experiments. All the approaches are listed next, including our proposed approaches.

(1) *TF Counting (TF).* We simply take out the k entities with the highest frequency in the target Web page p.
(2) *Supervised Learning (SL).* By training an SVM with the training pages labeled by human editors, we can obtain the relevant score for every entity in the target Web page p from SVM. Then we choose k entities with the largest score to be the result keywords [Yih et al. 2006]. This approach can be considered as a preprocessing step the of content- and advertisement-sensitive PageRank. The scores for the entities will be used as the content damping vectors of the iteration process of the content- and advertisement- sensitive PageRank.
(3) *Co-occurrence in Ads (CA).* We use a process to mine the co-occurrence of two entities in one advertisement content. Generally, if entity A and B occur in the same advertisement for more than t times, then we claim that A and B are friends. Given the target Web page p, we find the related entities to p with the most friends in p as the result.
(4) *Query Click-through Bipartite Gragh (QCBG).* Here we implement one traditional query expansion approach for the related Web page keywords recommendation. This approach is based on query clustering using both query content and the query-URL bipartite graph [Cao et al. 2008; Wen et al. 2001]. We use the AOL query click-through data to build the query-URL bipartite graph. After query clustering,

[11]with less than 100 terms.

we can generate an expansion for each target query. For the target Web page p, a candidate keyword set is obtained by the expansion on the seed keywords of p. Then, the closest k keywords to p are finally recommended, where the keywords are represented by vectors used in clustering and p is represented by merging the seed keywords vectors.

(5) *Content-sensitive PageRank (CPR).* We have proposed this approach in Section 4.4. As this algorithm does not use any advertisement information, it is used to compare the commercial impact with Algorithm 6.

(6) *Content- and Advertisement-sensitive PageRank (CAPR).* We have proposed this approach in Section 4.5. It is our key approach in this article.

5.3. Evaluation Measures

The input of the experiment is a target Web page p and the output is k keywords to p. For the gold-standard of the evaluation work, we invited five colleagues to judge the relevance of each page-keyword pair as follows.

— *Relevant and advertisable.* The keyword is relevant to the content of the target page and also it has a possibility to be valuable for advertising, scored as 1.
— *Otherwise.* The keyword is not considered as relevant to the content of the target page or it is impossible to be used as an advertising keyword, scored as 0.

Each page-keyword pair has at least two human judges. After the judgment work, we average the scores for each page-keyword pair. The scores can be interpreted as the possibility of relevance and advertisability[12]. Then we evaluate the performance of the algorithms using the evaluation measure described next.

In our experimental studies, we use P@n as the evaluation measure. Precision at position n (P@n) is defined to be the fraction of the top-n retrieved keywords that are relevant [Baeza-Yated and Ribeiro-Neto 2008].

$$P@n = \frac{\sum_{i=1}^{n} \tau_i}{n} \tag{7}$$

In Eq. (7), τ_i denotes the average rate score for the pair of the target Web page and the ith recommended keyword. Since we not only accept the good keywords occurring in p, but also the related ones, there is no good measure to evaluate the recall of each approach.

5.4. Experimental Results

Our experiments are divided into five parts. In the first part, we normally do the keywords recommendation with four approaches TF, SL, CPR, and CAPR. The output keywords can be both in-page keywords and leveraged keywords, called *universal keywords*. In the second part, we focus on the leveraged keywords recommendation (without in-page keywords in the result), using approaches implemented from algorithms CA, QCBG, CPR, and CAPR. The third part reports the impact of the parameters of CAPR. In the fourth part, we present a case study to analyze the ability of CAPR for dealing with the ambiguity of Wikipedia entities. In the last part, we demonstrate the recommendation performance against the word number of the target Web pages.

5.4.1. Universal Keywords Recommendation. Universal keywords recommendation judges the practical effectiveness of each algorithm, where both in-page and leveraged keywords are recommended. For every target Web page p, each approach provides a result of 20 keywords for p. We demonstrate the overall precision of the four approaches

[12]Term "advertisability" has been used in the work of Pandey et al. [2010].

Table III. Precision in Top k of the Results of the Four Approaches on Short-Text Web Page Set

	TF	SL	CPR	CAPR
Top5	0.4260	0.4195	0.5708	0.5751
Top10	0.3363	0.3248	0.5428	0.5580
Top15	0.2661	0.2653	0.4488	0.4753
Top20	0.2128	0.2203	0.4019	0.4514

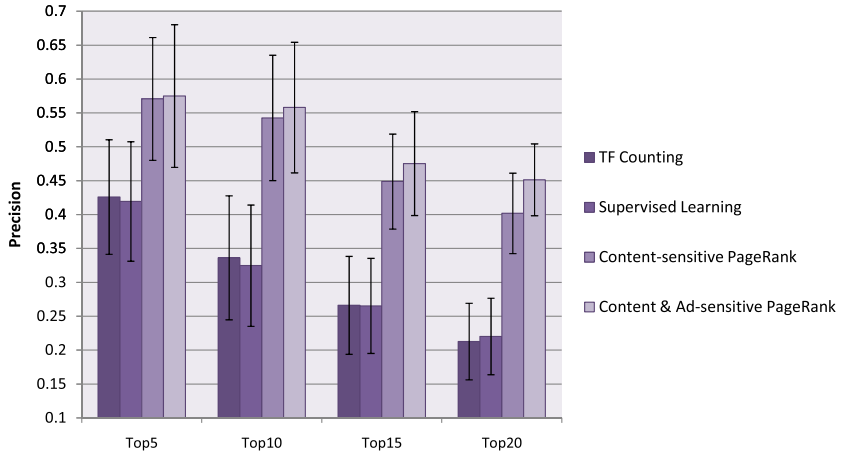

Fig. 5. Precision in top k of the results of the four approaches on short-text Web page set.

in top 5, top 10, top 15, and top 20 keywords recommended. Here we compare the results of the four approaches: TF, SL, CPR, and CAPR.

First we use the short-text Web page set for testing. The performance of the four approaches is shown in Table III and Figure 5.

Several observations are interesting to note: (i) CAPR shows significant improvement over other approaches on the short-text Web page set. Compared with SL, CAPR has an improvement of 37.09%, 71.82%, 79.15%, and 104.94% on top 5, 10, 15, and 20 respectively. It verifies that combining content- and advertisement-sensitive PageRank helps to find leveraged keywords which are more relevant to the target Web page than in-page keywords. (ii) Somewhat unexpectedly, the traditional supervised learning approach SL is no more effective than the TF counting approach, which is very simple and considered to be the baseline in the experiments. The possible reason is that SL over-emphasizes the page content and structure, which is sparse and very diverse in short-text Web pages. (iii) For most short-text target Web pages, the traditional approaches (TF, SL) could not even provide more than 15 keywords and the precision of the keywords is not so good, less than 45%. This reveals the problem of traditional keywords recommendation on short-text Web pages. In Section 5.4.5, we will give a panoramic view of the performance of these approaches against the size of target Web page. (iv) Compared with CPR, CAPR has an improvement of 0.76%, 2.81%, 5.90%, and 12.32% on top 5, 10, 15, and 20 respectively, which verifies the effectiveness of incorporating advertisement bias in CAPR.

Besides the short Web pages, we also demonstrate the performance of these approaches on overall Web pages set, not just short-text Web pages. The performance is shown in Table IV and Figure 6.

Table IV. Precision in Top k of the Results of the Four Approaches on Overall Web Pages Set

	TF	SL	CPR	CAPR
Top5	0.5782	0.6356	0.6622	0.6693
Top10	0.4792	0.4921	0.5443	0.5624
Top15	0.4238	0.4284	0.4897	0.5149
Top20	0.3693	0.3822	0.4622	0.4965

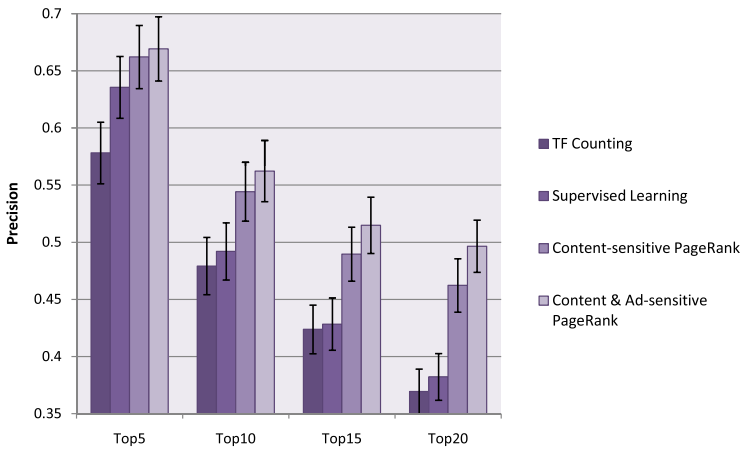

Fig. 6. Precision in top k of the results of the four approaches on overall Web pages set.

As is shown in Figure 6, CAPR performs the best in all the comparisons and has an improvement of 5.30%, 14.29%, 20.18%, and 29.92% on top 5, 10, 15, and 20 over SL respectively. It proves that CAPR also works well on the long-text Web pages even though the improvement is not so significant as on short-text Web pages and advertisement bias also helps improve the performance. Moreover, SL performs better than the baseline TF, which indicates that SL works well mainly on long-text Web pages. Compared with those on short-text Web pages, the performance of each algorithm has lower standard error, shown as error bars.

Since CAPR recommends both in-page keywords and leveraged keywords, we also did an analysis of the in-page keywords proportion in the recommendation of CAPR, as is shown in Table V and Figure 7. From the result we have following observations: (i) As the number of recommended keyword increases, the in-page keyword proportion decreases. For each target Web page, the in-page keyword number has an upper limit. Thus after the most suitable in-page keywords are recommended, more and more leveraged keywords are more likely to be recommended than the remaining in-page keywords. (ii) On overall Web pages set, the in-page proportion is smaller than that in the overall Web pages set. This is also reasonable because there are fewer in-page keywords in the short-text Web pages and leveraged keywords are more likely to occur in the result.

As a case study of the comparison for the approaches of TF, SL, and CAPR, we here provide their performance on a specific short-text test Web page case of *ION Media Networks*[13]. The performance is shown in Table VI. From the performance result,

[13]ION Media Networks. http://www.ionmedia.tv/.

Table V. Proportion of In-Page Keywords in the Recommendation of Content- and Advertisement-Sensitive PageRank on Overall Web Pages Set and Short-Text Web Page Set

	Overall Web Page Set	Short Web Page Set
Top5	0.9089	0.6752
Top10	0.7366	0.5373
Top15	0.6640	0.4374
Top20	0.6279	0.4125

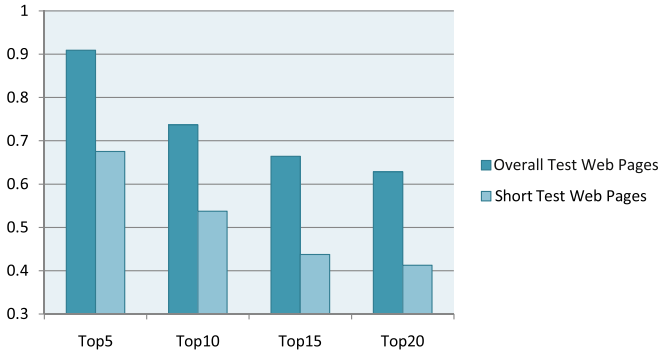

Fig. 7. Proportion of in-page keywords in the recommendation of content- and advertisement-sensitive PageRank on overall Web pages set and short-text Web page set.

Table VI. 20 Keywords Recommended to Home Page of *ION Media Networks*

	TF	SL	CAPR
1	**ION Media Networks**	**ION Media Networks**	**ION Media Networks**
2	**Television**	**ION Life**	**ION Life**
3	**ION Life**	**Television**	**Television**
4	Company	power	power
5	Life	Completed	Completed
6	Completed	Company	**Broadcasting**
7	power	Life	Company
8	**Sony Pictures Television**	CBS	**Comcast**
9	Twentieth Television	DTV	Qubo
10	RHI Entertainment	Bros	**NBC**
11	Entertainment	Video	Rock music
12	**NBC Universal**	Pictures	Buffalo, New York
13	Open Mobile	Coalition	**American Broadcasting**
14	Coalition	Universal	Touched by an Angel
14	Universal	Entertainment	**Fox Sports Net**
16	Pictures	**Sony Pictures Television**	**Talk radio**
17	Video	**Twentieth Television**	**Ion Television**
18	Bros	RHI Entertainment	**Cornerstone Television**
19	CBS	**NBC Universal**	Rochester, New York
20	DTV	Open Mobile	Wilmington, Delaware
Hit	6	6	11

we can know that TF hits 6 keywords, SL hits 6 keywords, and CAPR hits up to 11 keywords in the top 20, which dominates in the case study.

5.4.2. Leveraged Keywords Recommendation. Here we investigate the performance only on the recommended leveraged keywords. Among the compared algorithms, CA,

Table VII. Precision in Top k of the Results of the Four Approaches on Short-Text Web Page Set

	CA	QCBG	CPR	CAPR
Top5	0.3025	0.4320	0.5027	0.5033
Top10	0.1992	0.3430	0.4367	0.4534
Top15	0.1470	0.2967	0.4210	0.4403
Top20	0.1219	0.2680	0.3696	0.4119

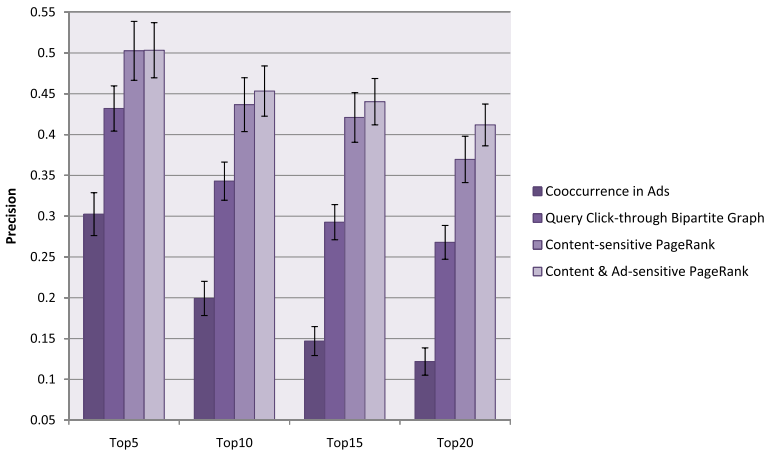

Fig. 8. Precision in top k of the results of the four approaches on short-text Web page set.

QCBG, CPR, and CAPR have the ability to recommend leveraged keywords of the target Web pages, simply by filtering out the in-page keywords in the results. Similar to Section 5.4.1, we first demonstrate the precision of the four approaches on top 5, 10, 15, and 20 keywords recommendation on the short-text Web page set. The performance of those approaches is shown in Table VII and Figure 8.

From the results, we can know that: (i) CAPR gives a high performance in the work of leveraged keywords recommendation. It has an improvement of 16.52%, 32.18%, 50.46%, and 53.69% on top 5, 10, 15, and 20 against QCBG. Its performance of leveraged keywords precision could even compare to the universal keywords recommendation performance. (ii) On the other hand, CA is not as suitable for related advertising keywords recommendation. For one reason, the short-text Web page makes it much more difficult to determine the topic or the keywords from which to get the leveraged keywords; for another reason, there is so much noise in the advertisement text, such as misleading and ambiguous expressions in the advertisement text and the unbalance of the entities' frequency in the advertisement set, which misguides the relation of two entities in the mining process of co-occurrence in ads. (iii) QCBG performs a little worse than CAPR on the top 5 results. However, as the recommended keyword number increases, the performance gap between these two algorithms increases significantly. This is because the user queries are highly diverse and thus query expansion will possibly import some irrelevant keywords. (iv) Furthermore, the result also demonstrates that advertisement bias helps improve the performance of leveraged keywords by 0.53%, 3.82%, 5.59%, and 11.43% on top 5, 10, 15, and 20. From the performance figures, we can conclude that the ability to find leveraged keywords is very important in the field of advertising keywords recommendation, especially on the short-text Web page set.

Table VIII. Precision in Top k of the Results of the Four Approaches on Overall Web Page Set

	CA	QCBG	CPR	CAPR
Top5	0.2080	0.3980	0.5375	0.5440
Top10	0.1813	0.3178	0.4938	0.5438
Top15	0.1686	0.2891	0.4392	0.5137
Top20	0.1611	0.2772	0.4167	0.4750

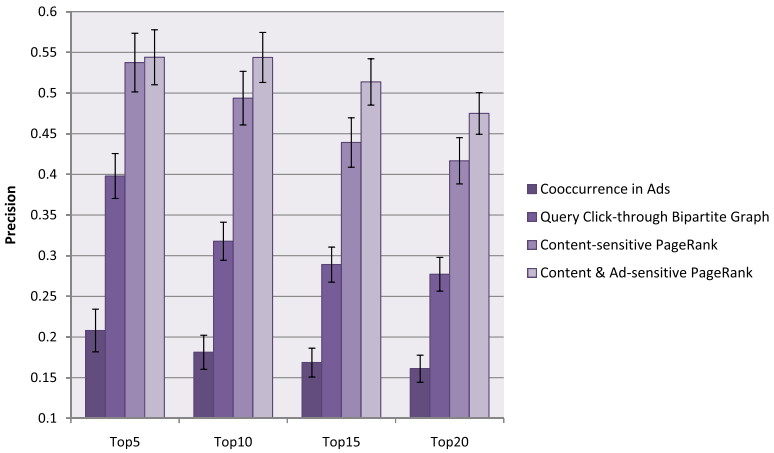

Fig. 9. Precision in top k of the results of the four approaches on overall Web page set.

We also make a universal keywords recommendation for comparison on an overall Web page set, not just short-text Web pages. The performance is shown in Table VIII and Figure 9.

As is shown in Table VIII, CAPR dominates in each comparison and has an improvement of 36.67%, 71.10%, 77.69%, and 71.34% on top 5, 10, 15, and 20 over QCBG respectively. From the result we can know CAPR still works well on the long Web pages and advertisement bias also helps improve the performance by 1.21%, 10.14%, 16.96%, and 13.99% on top 5, 10, 15, and 20 respectively. Thus it verifies that CAPR is the most competent to recommend leveraged keywords to Web pages.

5.4.3. Parameter Tuning. As is shown in Eq. (5), there are two parameters, α and β, in CAPR that need to be tuned. In this section, we focus on the impact of the parameters on the performance of the algorithm.

In our experiment, the two parameters are tuned separately. First, we fix $beta$ to 5.0×10^{-5} and tune α from 0.30 to 0.98 (some preliminary experiments have shown that this area produces the best result). Secondly, we fix α to 0.9, which is sensitive to our Wikipedia graph, and tune β from 1.0×10^{-5} to 1.0×10^{-3}. The precision on the top 20 recommended keywords can be depicted against α and β, which is shown in Figure 10.

From Figure 10, it can be noted that there is a trade-off between the weight of content bias and the propagation of PageRank by tuning α and when β is larger than 2.0×10^{-5}, the performance reduces as β increases. Finally, we set $\alpha = 0.85$ and $\beta = 1.5 \times 10^{-5}$ to run CAPR. It is necessary to point out that α and β here have already been combined with the normalization factor of the respective damping vectors, which explains the gap between the orders of magnitude α and β.

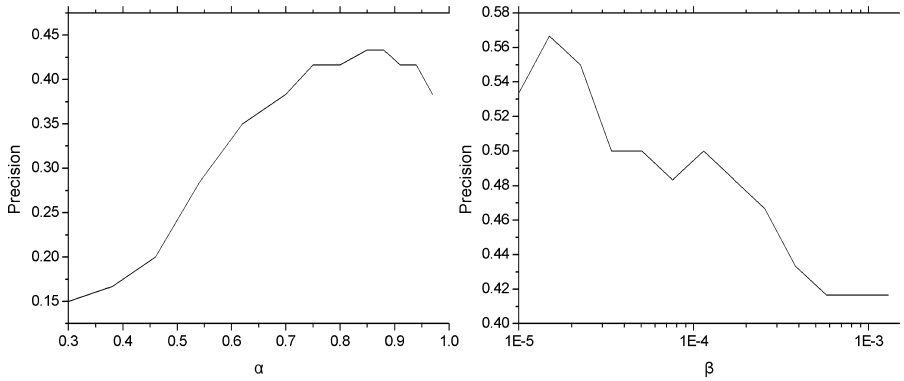

Fig. 10. Precision in top 20 of the results against the parameters α and β.

Table IX. Top 20 Keywords Offered by CAPR from the Seed Entites: *Apple*, *IPod*, and *IPhone*, where Apple is a Noise Seed Entity

Seed entities: Apple, IPod, IPhone		
Result Top 20 Keywords:		
IPhone	**IPod**	Apple
E-mail	**Comparison of iPod managers**	**Apple Inc.**
Mac OS X	**ITunes**	**Steve Jobs**
IPod Touch	United States Dollar	Wired (magazine)
Macworld	**IPhone OS**	**Bluetooth**
ITunes Store	**Nokia**	**Flash memory**
Portable media player	**ARM architecture**	

5.4.4. Noise Impact and Ambiguity Analysis. As has been discussed in Section 4.2, the seed entities of the propagation step are given by traditional recommendation approaches. Will the propagation step of CAPR amplify the errors made by these approaches if there are some irrelevant keywords, called *noise*, in the seed entity list? Moreover, in the application of Wikipedia, one common problem is the ambiguity of entity names. In the area of keywords recommendation, this problem still exists. For example, if a term *Apple* occurs in the target Web page, there is a question as to whether it means the fruit or the Apple brand[14]. In Wikipedia, the term *Apple* will directly be matched to the one under the fruit category. So if the term *Apple* in the page actually means the brand of Apple, the ambiguity problem exists.

However, from the preceding experiments, it is found that the noise and ambiguity problems do not seem to sinificantly reduce the performance of CAPR. We claim that topic-sensitive PageRank can reduce the impact of noise and ambiguity of Wikipedia entities. Here we provide a preliminary analysis. Given some seed entities, several of which may be noisey or ambiguous and can match different Wikipeida categories. Since we have removed the cross-category edges in the graph, the ambiguous entities cannot distribute much PageRank value to their neighbors unless the ambiguous entities outnumber the entities under the correct first-level category. For a concise example, for a target Web page about the electronic product of Apple, we are given three seed entities: *Apple, IPod, and IPhone*, where *Apple*, as is mentioned before, is a noise in the seed entities. We present the top 20 recommended keywords from CAPR in Table IX.

[14]Apple Inc. http://www.apple.com/.

Table X. Precision in Top 10 of the Results of the Three Approaches against the Page Size

Page Size	TF	SL	CAPR
0 to 99	0.3363	0.32.48	0.5580
100 to 399	0.5647	0.6294	0.6588
400 to 799	0.5778	0.6815	0.6889
800 +	0.6571	0.6786	0.7357

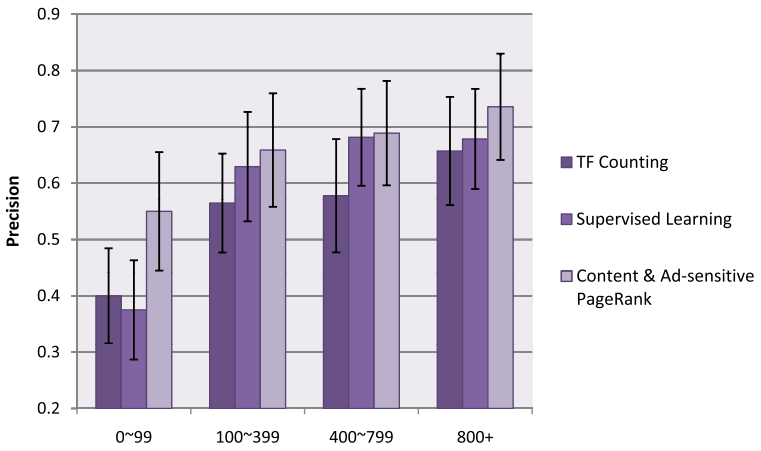

Fig. 11. Precision in top 10 of the results of the three approaches against the page size.

From Table IX we can see that 17 keywords are in the topic of Electronic Products, 2 keywords are Wikipedia noise entities, and only *Apple* belongs to the fruit topic. This example verifies the disambiguating ability of content- and advertisement-sensitive PageRank.

5.4.5. Performance against Target Web Page Content Size. In the last part of our experiment, we analyze the performance against the content size of target Web pages. We make a comparison on TF, SL, and CAPR on the page size domain of 1 to 15104. Specifically, we divide the pages into 4 groups by their word numbers, the intervals of which are $1 \sim 99$, $100 \sim 399$, $400 \sim 799$, and 800+. The performance on the top 10 keywords recommended of three algorithms on each group of pages is in Table X and Figure 11.

From the result we observe the following. (i) On average, the performance of keywords recommendation improves as the content size of the target Web page increases while when the page content size is especially small, for example, less than 100, the traditional approaches (TF and SL) for keywords recommendation do not work well (less than 40% in our experiment). (ii) Our approach CAPR still works well on the short-text pages (55.01% in our experiment), making an improvement of 91.30% on SL. (iii) As the content size increases, traditional approaches show large variation (SL varies from 37.50% to 67.86%, increasing by 80.96%) but our algorithm indicates more robustness and less sensitivity to the content size of the target Web pages (from 55.01% to 73.57%, increasing by 33.74%).

5.5. Discussion

Based on the five parts of the experiment, we claim that content- and advertisement-sensitive PageRank works well for advertising keywords recommendation. It provides

a significant improvement over several state-of-the-art approaches on short-text Web pages. For short-text target Web pages, the keywords recommended by traditional approaches are always with noise. However, in the propagation step of our approach, the noise is reduced and the keywords in the main topics get higher ranks due to the propagation of the content- and advertisement-sensitive PageRank score. In addition, leveraged keywords which do not occur in but are still relevant to the target Web page are also recommended. As a result, the problems of poor content, simple structure, and lack of candidate keywords for short-text Web pages are solved.

6. CONCLUSION AND FUTURE WORK

Traditional approaches depending on the abundance of textual information in the target Web pages do not work well in the context of short-text Web pages. To address this important problem, we propose a novel approach using content- and advertisement-sensitive PageRank on the Wikipedia graph. In the experiment, our approach yields a high improvement over traditional approaches in the precision of top 20 keywords on short-text target Web pages. It verifies that content- and advertisement-sensitive PageRank is an effective approach to advertising keywords recommendation on short-text Web pages.

In the future work, we plan to refine the Wikipedia graph, such as refining the edges to be more precise to identify the semantic similarity between two entities. For the efficiency improvement of our system, we will implement parallelization. As the WikiGraph has been divided into κ subgraphs based on the category, the computation on these subgraphs can be done in parallel. Furthermore, more parallelism can be achieved using more sophisticated technology of distributed computation for each subgraph. For other applications, we can change the PageRank bias into user profile information and implement our algorithm in personalized search. In addition, our algorithm can be adapted to product recommendation on a Web store with the interlinked product pages.

REFERENCES

ABHISHEK, V. AND HOSANAGAR, K. 2007. Keyword generation for search engine advertising using semantic similarity between terms. In *Proceedings of the 9th International Conference on Electronic Commerce*. 89–94.

ANAGNOSTOPOULOS, A., BRODER, A., GABRILOVICH, E., JOSIFOVSKI, V., AND RIEDEL, L. 2007. Just-in-time contextual advertising. In *Proceedings of the CIKM Conference*.

ANTONELLIS, I., GARCIA-MOLINA, H., AND CHANG, C.-C. 2008. Simrank++: Query rewriting through link analysis of the click graph. In *Proceedings of the International Conference on Very Large Databases (VLDB)*. 408–421.

BAEZA-YATED, R. AND RIBEIRO-NETO, B. 2008. *Modem Information Retrieval*. Addison-Wesley Longman Publishing Co., Boston, MA.

BECKER, H., BRODER, A., GABRILOVICH, E., JOSIFOVSKI, V., AND PANG, B. 2009. What happens after an ad click? quantifying the impact of landing pages in web advertising. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*. 57–66.

BOLDI, P., BONCHI, F., CASTILLO, C., DONATO, D., GIONIS, A., AND VIGNA, S. 2008. The query-flow graph: Model and applications. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM)*. 609–618.

BOLDI, P., BONCHI, F., CASTILLO, C., DONATO, D., AND VIGNA, S. 2009. Query suggestions using query-flow graphs. In *Proceedings of the Workshop on Web Search Click Data (WSCD)*. 56–63.

BRIN, S. AND PAGE, L. 1997. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the 7th International World Wide Web Conference*. 107–117.

BRIN, S., MOTWANI, R., PAGE, L., AND WINOGRAD., T. 1998. What can you do with a web in your pocket. *Bull. IEEE*.

BRODER, A., FONTOURA, M., JOSIFOVSKI, V., AND RIEDEL, L. 2007. A semantic approach to contextual advertising. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information*. 559–566.

BRODER, A. Z., CICCOLO, P., FONTOURA, M., GABRILOVICH, E., JOSIFOVSKI, V., AND RIEDEL, L. 2008. Search advertising using web relevance feedback. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM)*. New York, NY, 1013–1022.

BRODER, A., CICCOLO, P., GABRILOVICH, E., JOSIFOVSKI, V., METZLER, D., RIEDEL, L., AND YUAN, J. 2009. Online expansion of rare queries for sponsored search. In *Proceedings of the 18th International Conference on World Wide Web*.

CAO, H., JIANG, D., PEI, J., HE, Q., LIAO, Z., CHEN, E., AND LI, H. 2008. Context-aware query suggestion by mining click-through and session data. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 875–883.

CARMEL, D., ROITMAN, H., AND ZWERDLING, N. 2009. Enhancing cluster labeling using wikipidia. In *Proceedings of 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 139–146.

CHEN, Y., XUE, G., AND YU, Y. 2008. Advertising keyword suggestion based on concept hierarchy. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining*. 251–260.

CHOI, Y., FONTOURA, M., GABRILOVICH, E., JOSIFOVSKI, V., MEDIANO, M., AND PANG, B. 2010. Using landing pages for sponsored search ad selection. In *Proceedings of the 19th International Conference on World Wide Web*. 251–260.

CRISTO, M., RIBEIRO-NETO1, B., GOLGHER, P. B., AND DE MOURA, E. 2006. Search advertising. In *Proceedings of the StudFuzz Conference 197*. 259–285.

FANG, Y., WU, B., LI, Q., BOT, R., AND CHEN, X. 2005. Domain-specific keyphrase extraction. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*. 283–284.

FENG, J., BHARGAVA, H., AND PENNOCK, D. 2003. Comparison of allocation rules for paid placement advertising in search engines. In *Proceedings of the 5th International Conference on Electronic Commerce*. 294–299.

HAVELIWALA, T. 2002. Topic-Sensitive pagerank. In *Proceedings of the 14th World Wide Web Conference*. 517–526.

HILLARD, D., SCHROEDL, S., MANAVOGLU, E., RAGHAVAN, H., AND LEGGETTER, C. 2010. Improving ad relevance in sponsored search. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining*. 361–369.

HU, J., FANG, L., CAO, Y., ZENG, H.-J., LI, H., YANG, Q., AND CHEN, Z. 2008. Enhancing text clustering by leveraging wikipedia semantics. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 179–186.

HU, J., WANG, G., LOCHOVSKY, F., SUN, J., AND CHEN, Z. 2009. Understanding user's query intent with wikipedia. In *Proceedings of the 18th World Wide Web Conference*. 471–478.

IAB AND PRICEWATERHOUSECOOPERS. 2011. `http://www.iab.net/media/file/IAB\_Full\_year\_2010\_0413\_Final.pdf`.

JONES, R., REY, B., MADANI, O., AND GREINER, W. 2006. Generating query substitutions. In *Proceedings of the 15th International Conference on World Wide Web*. 387–396.

JONES, S. AND PAYNTER, G. 2001. Human evaluation of kea, an automatic keyphrasing system. In *Proceedings of the 1st ACM/IEE-CS Joint Conference on Digital Libraries*. 148–156.

JOSHI, A. AND MOTWANI, R. 2006. Keyword generation for search engine advertising. In *Proceedings of the 6th IEEE International Conference on Data Mining (Workshops)*.

KLEINBERG, J. M. 1999. Authoritative sources in a hyperlinked environment. *J. ACM 46*, 604–632.

LITVAK, M. AND LAST, M. 2008. Graph-based keyword extraction for single-document summarization. In *Proceedings of the Workshop on Multisource Multilingual Information Extraction and Summarization (Coling)*. 17–24.

MATSUO, Y. 2003. Keyword extraction from a single document using word co-occurrence statistical information. *Int. J. Artif. Intell. Tools*.

MEDELYAN, O., MILNE, D., LEGG, C., AND WITTEN, I. 2009. Mining meaning from wikipedia. *Int. J. Hum.-Comput. Studies.*, 716–754.

MITRA, M., SINGHAL, A., AND BUCKLEY, C. 1998. Improving automatic query expansion. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 206–214.

PAGE, L. 1997. Pagerank: Bringing order to the web. In *Digital Libraries Working Paper*.

PANDEY, S., PUNERA, K., FONTOURA, M., AND JOSIFOVSKI, V. 2010. Estimating advertisability of tail queries for sponsored search. http://arnetminer.org/viewpub.do?pid=2814327.

RADLINSKI, F., BRODER, A., CICCOLO, P., GABRILOVICH, E., JOSIFOVSKI, V., AND RIEDEL, L. 2008. Optimizing relevance and revenue in ad search: A query substitution approach. In *Proceedings of the 31st International ACM SIGIR Conference on Research and Development in Information Retrieval*. 403–410.

RAGHAVAN, H. AND HILLARD, D. 2009. A relevance model based filter for improving ad quality. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 762–763.

RAVI, S., BRODER, A., GABRILOVICH, E., JOSIFOVSKI, V., PANDEY, S., AND PANG, B. 2010. Automatic generation of bid phrases for online advertising. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining*. 341–350.

RIBEIRO-NETO, B., CRISTO, M., GOLGHER, P., AND MOURA., E. 2005. Impedance coupling in content-targeted advertising. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 496–503.

SCHÖNHOFEN, P. 2006. Identifying document topics using the wikipedia category network. *Web Intell. Agent Syst.,* 456–462.

SWENEY, M. 2009. http://www.guardian.co.uk/media/2009/sep/30/internet-biggest-uk-advertising-sector.

TURNEY, P. D. 2000. Learning algorithms for keyphrase extraction. *J. Inform. Retrieval*., 303–336.

TURNEY, P. D. 2003. Coherent keyphrase extraction via web mining. In *Proceedings of the IJCAI'03 Conference*. 434–439.

WANG, C., ZHANG, P., CHOI, R., AND EREDITA, M. 2002. Understanding consumers attitude toward advertising. In *Proceedings of the 8th Americas Conference on Informatino System*. 1143–1148.

WANG, H., LING, Y., FU, L., XUE, G., AND YU, Y. 2009a. Efficient query expansion for advertisement search. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 51–58.

WANG, P. AND DOMENICONI, C. 2008. Building semantic kernels for text classification using wikipedia. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 713–721.

WANG, P., HU, J., ZENG, H.-J., AND CHEN, Z. 2009b. Using wikipedia knowledge to improve text classification. *Knowl. Inf. Syst. 19*, 265–281.

WEN, J.-R., NIE, J.-Y., AND ZHANG, H.-J. 2001. Clustering user queries of a search engine. In *Proceedings of the 10th International Conference on World Wide Web (WWW)*. 162–168.

WITTEN, I., PAYNTER, G., FRANK, E., GUTWIN, C., AND NEVILL-MANNING, C. 1999. Kea: Practical automatic keyphrase extraction. In *Proceedings of the 4th ACM Conference on Digital Libraries*. 254–255.

YIH, W., GOODMAN, J., AND CARVALHO, V. 2006. Finding advertising keywords on web pages. In *Proceedings of the 15th World Wide Web Conference*. 213–222.

YU, J., THOM, J., AND TAM, A. 2007. Ontology evaluation using wikipedia categories for browsing. In *Proceedings of the 6th ACM Conference on Information and Knowledge Management*. 223–232.

Received April 2011; revised September 2011; accepted October 2011