Selecting Keywords to Represent Web Pages Using Wikipedia Information

Maisa Vidal
Univ. Fed. do Amazonas
Instituto de Computação
Manaus, AM, Brasil
maisa@icomp.ufam.edu.br

Guilherme V. Menezes
Univ. Fed. de Minas Gerais
Dep. de Ciência da Comput.
Belo Horizonte, MG, Brasil
gmenezes@dcc.ufmg.br

Klessius Berlt
Univ. Fed. do Amazonas
Instituto de Computação
Manaus, AM, Brasil
klessius@icomp.ufam.edu.br

Edleno S. de Moura
Univ. Fed. do Amazonas
Instituto de Computação
Manaus, AM, Brasil
edleno@icomp.ufam.edu.br

Karla S. O. Gomes
Intituto Nokia de Tecnologia
Manaus, AM, Brasil
karla.gomes@indt.org.br

Nivio Ziviani
Univ. Fed. de Minas Gerais
Dep. de Ciência da Comput.
Belo Horizonte, MG, Brasil
nivio@dcc.ufmg.br

David Fernandes
Univ. Fed. do Amazonas
Instituto de Computação
Manaus, AM, Brasil
david@icomp.ufam.edu.br

Marco Cristo
Univ. Fed. do Amazonas
Instituto de Computação
Manaus, AM, Brasil
marco.cristo@icomp.ufam.edu.br

ABSTRACT

In this paper we present three new methods to extract keywords from web pages using Wikipedia as an external source of information. The information used from Wikipedia includes the titles of articles, co-occurrence of keywords and categories associated with each Wikipedia definition. We compare our methods with three keyword extraction methods used as baselines: (i) all the terms of a web page, (ii) a TF-IDF implementation that extracts single weighted words of a web page and (iii) a previously proposed Wikipedia-based keyword extraction method presented in the literature. We compare our three keyword extraction methods with the baseline methods in three distinct scenarios, all related to our target application, which is the selection of ads in a context-based advertising system. In the first scenario, the target pages to place ads were extracted from Wikipedia articles, whereas the target pages in the other two scenarios were extracted from a news web site. Experimental results show that our methods are quite competitive solutions for the task of selecting good keywords to represent target web pages, albeit being simple, effective and time efficient. For instance, in the first scenario our best method used to extract keywords from Wikipedia articles achieved an improvement of 33% when compared to the second best baseline, and a gain of 26% when considering all the terms.

Webmedia '2012 São Paulo, SP, Brasil

Categories and Subject Descriptors

H. Information Systems [**H.m. Miscellaneous**]: Databases

Keywords

Context-based advertising, keyword extraction, Wikipedia

1. INTRODUCTION

Context-based advertising has become widespread in the web in recent years due to its success in generating revenue to a large variety of web sites, ranging from small blogs to large online newspapers. A system that implements this form of advertising works by automatically associating ads with the content of the web site a user is currently visiting. The general assumption is that users have a higher probability of finding an ad interesting if the content of the ad is similar to the content of the web page the user is currently browsing in.

Traditional context-based advertising systems usually work in two phases: (i) they represent the contextual information in the target web page, e.g., by extracting the keywords that best summarize its content; (ii) they match the representations of the target page and each ad of the ad inventory database, building a ordered list of the ads to be displayed to the users. The order of the resulting ad list is determined, among other elements, by the similarity between the contents of the page and the ad.

In this paper we focus in the first phase, i.e., extraction of keywords from target web pages. We present new alternatives for representing the contextual information present in a web page. We use information from Wikipedia articles to semantically enrich the information that best summarize web pages.

Wikipedia is a source of information with some interesting characteristics. First, it is very large and covers many knowledge domains. At the time we performed our experiments, Wikipedia had approximately 3.3 million articles in

English. Second, due to its popularity and the open nature of its editing process, it is frequently updated providing fresh content about many topics. As a consequence of such broader and dynamic nature, it is likely to have detailed information on new products that draw public attention, such as laptop models, game consoles or recently published books. This is very convenient to context-based advertising systems, since the user context and their interests also have a dynamic nature, that is, they change over months, weeks, and even during the same day. Further, each Wikipedia article describes an entity, such as a place, a person or an object, and is semantically precise in the sense that each article describes a "unit of knowledge" in an unambiguous way.

To take advantage of Wikipedia as a source of knowledge useful for context-based advertising systems, we propose three methods that use information from Wikipedia to represent web pages. The first one uses the titles of Wikipedia articles as a controlled vocabulary for extraction of semantically-rich keywords from target pages, using a TF-IDF weight scheme. The other two methods use the category of Wikipedia article to expand the representation of the web pages by assigning keywords that are not present in the pages, but are related to their categories.

The three methods were engineered to be simple and present small computational costs. Thus, the proposed methods are suitable for industrial context-based advertising systems that are required to process millions of requests every day. In these systems, every page view in the web site generates a request, and any method that needs too much computational effort would not be acceptable.

We evaluate the effectiveness of our proposed methods by comparing them to three keyword extraction baselines: (i) a method considering all terms of the target page; (ii) a TF-IDF implementation to select single words; (iii) a popular Wikipedia-based keyword extraction algorithm described in [12].

Experimental results show that the proposed methods are competitive in practice. For instance, when selecting keywords from Wikipedia articles, our best method outperformed the representations based on all the terms (i) and TF-IDF weighting (iii) with gains of about 33% and 26 %, respectively. In the worst scenario we found in our experiments, our methods achieved results similar to the approach proposed by [12].

This paper is structured as follows. In Section 2, we present the related work. We present the three proposed methods in Section 3. We present experimental results comparing our three methods with the three baseline methods in Section 4. Finally, in Section 5, we present our conclusions and review our contributions.

2. RELATED WORK

Wikipedia has been used in some previous work as a measure of semantic relatedness. In [3], the authors introduce a method called Explicit Semantic Analysis (ESA), with the objective of semantically enriching texts in natural language. It represents texts in a vector of Wikipedia concepts, associating a weight to each concept quantifying the relatedness between the text and the concept. They experimented with word-level and text-level semantic relatedness, obtaining better results than other approaches in the literature. They also experimented the effectiveness of their semantic relatedness measure for generating features for classification. While their focus was to address classification problems, the positive results obtained are useful to illustrate the potential benefits of using Wikipedia information to better understand and represent the content of web pages.

Hu et al [6] employ a Wikipedia based semantic relatedness measure to improve the effectiveness of document clustering. First, the method maps the text of a document to Wikipedia concepts using an algorithm to disambiguate possible candidate concepts. Next, it calculates a similarity value based on the category hierarchy of the concepts that annotate the document. It also explores Wikipedia redirects and disambiguation pages to obtain a broader representation of each concept and improve the matching. They obtained significant improvements when compared to [3].

In [8] the authors also enrich textual documents with Wikipedia concepts to improve clustering. The method maps a text to concepts by leveraging the collection of anchor texts related to Wikipedia articles, i.e., they use the anchor text of links that point to a concept to describe this concept. Each occurrence of an anchor text in the document maps the document to a candidate concept. The set of candidate concepts is subsequently reduced by using a cluster-based feature selection algorithm. A follow-up work from the same authors is presented in [9], in which they compare their method with other related methods in the literature, including [3] and [6].

In [7] the authors devised another method of Wikipedia semantic relatedness that associate texts with Wikipedia categories. They experimented with two matching techniques: exact matching and relatedness matching. The first technique performs exact string matching, which is very efficient, while the second uses the cosine measure to select concepts. Interestingly, exact matching has performed better in most cases, showing that it represented the best alternative to match textual documents and concepts. Additional applications for Wikipedia semantic relatedness include cluster labeling [2] and search [11]. All the methods mentioned above have used information available in Wikipedia to improve classification or clustering of documents. We here are interested in use Wikipedia information as a keyword extraction method.

Among the keyword extraction methods that use Wikipedia information, we can cite [12], which uses information from Wikipedia links to extract keywords. The idea behind the method is to estimate the probability that a phrase is a keyword by calculating how many times the phrase was linked to other Wikipedia articles. Another example of method that uses Wikipedia information to extract keywords is [5], which uses Wikipedia to create a semantic graph between the terms of a Web page. They use the property that keywords related to the main topic of the document usually belong to the most interconnected communities in the term graph. The work by [12] was selected to be included in our experiments to help the reader to compare our method with the other methods proposed. It gives good results in practice and has computational costs close to our proposal.

The authors in [15] introduce WikiRank, a graph-based method that extracts key Wikipedia concepts from a text. The method explores the hyperlink connections among Wikipedia articles and categories to obtain a ranking of related concepts. They analyze their method in two applications. The first one is a Wikipedia concept linking application, which shows Wikipedia definitions related to keywords in a news

text. The second one is a visualizer of the key concepts in a news collection.

The methods presented by Mihalcea and Csomai [12], Grineva et al [5], and Zhou et al [15] are all proposals to extract keywords from web pages using Wikipedia information. The main difference to our proposal is that these three works adopt graph-based strategies to extract keywords from pages, while our methods use occurrence statistics about Wikipedia titles and information about the category these titles belong to. Experimental results show that our methods outperform the work presented in [12] in the target application, which is associate ads and product offers with web pages. We do not provide a comparison to the other two methods due to the lack of detailed information about them. We also found no comparison between the previously proposed methods, so we were not able to even compare our proposal to the other two found in the literature.

Some previously proposed methods use machine learning techniques to select keywords from a given web page. Irmak et al [10] use click-through data to train a model to rank a list of pre-determined entities of a document according to their interestingness and relevance. Goodman and Carvalho [4] and Yih et al [14] use logistic regression to learn good keywords for advertising and also study a large number of features to determine the importance of a keyword. Although these methods present good results on selecting keywords, the difference to our approach is that they require an extra effort to perform the training step. Also note that our method is complementary to these ones since our results can be used as additional feature by the machine learning strategies proposed by those authors.

3. WIKIPEDIA-BASED KEYWORD EXTRACTION

In this section we present three new keyword extraction methods that use Wikipedia semantic information. The main objective of the proposed methods is to improve the matching between ads and target web pages.

3.1 Wiki-TF-IDF

The first method for keyword extraction uses the titles of Wikipedia articles as a controlled vocabulary to extract keywords from target web pages. The method uses a variant of the TF-IDF weighting scheme [1], which we refer to as Wiki-TF-IDF method. The terms belonging to the titles of Wikipedia articles are used to obtain their term frequency (TF) in the target page and their inverse document frequency (IDF) in Wikipedia titles. The two weights TF-IDF are used to rank the candidate keywords. The intuition behind the method is that only semantically rich units of information are extracted, thus reducing noise and ambiguity in the extracted set of keywords.

By extracting more accurate keywords, Wiki-TF-IDF improves the matching to external datasets. For example, consider that phrase "New York" is present in a target web page. If we take words as semantic units of information, the keywords "New" and "York" are extracted. The word ("New") carries very low information value, and might be considered as a stopword which would be removed by some methods. The word ("York") is ambiguous, since it may refer to the city of York or to the New York city. Since "New York" is the title of an article in Wikipedia, our method considers the phrase "New York" as a single semantic unit, avoiding the negative effect of noisy keywords.

The Wiki-TF-IDF algorithm has two steps:

1. It extracts candidate keywords from the target web page, considering that: (i) It matches the largest possible phrase present in a text fragment. For instance, in the phrase "World Wide Web" it only considers the full phrase as a candidate, discarding the subphrases "World Wide", "Wide Web", "World", "Wide", and "Web". (ii) The keywords that occur less then 3 times in the collection of Wikipedia articles are excluded. (iii) Only fragments of, at most, eight words are considered. Thus, the largest keyword to be considered has eight words, thus limiting the time taken to extract them.

2. It ranks the candidate keywords using the following TF-IDF scheme[1]:

$$w_{ij} = (1 + \log(f_{ij})) * \log(\frac{N}{n_i}) \quad (1)$$

where f_{ij} is the frequency of the candidate keyword c_i in document d_j, N is the number of documents in Wikipedia, and n_i is the number of Wikipedia documents in which the candidate c_i occurs at least once. Then, it ranks the candidate keywords and the top k keywords are selected.

Wiki-TF-IDF is expected to perform efficiently even in large-scale systems with a large number of users. Its efficiency is due to (i) its simplicity and (ii) its use of a hash structure to store Wikipedia titles which allows it to extract all the candidate keywords in linear time.

3.2 Wiki-Categories-1

The second method for keyword extraction considers the categories that each Wikipedia article belongs to, which we refer to as Wiki-Categories-1. The Wiki-Categories-1 algorithm has three steps:

1. It selects from a given web page P a set of initial keywords IKw_P with the top 10 keywords ranked with the Wiki-TF-IDF method. The method could in fact select an initial set of any size, but when analyzing initial results provided by method Wiki-TF-IDF, we realized that it usually achieves a good precision when selecting 10 keywords, and further, the quality of results usually does not improve much when selecting more than 10 keywords.

2. Then, it identifies the set of categories in Wikipedia related to the keywords of IKw_P, creating a new set of keywords CKw_P that contains all the categories related to the titles found in IKw_P. Remember that each keyword selected by Wiki-TF-IDF is in fact a title of a Wikipedia article. Further, each Wikipedia title belongs to one or more categories in Wikipedia. Finally, each Wikipedia article presents a list of categories it belongs to.

3. The title of the categories identified in CKw_P are all listed in a single text that is then sent as input to the Wiki-TF-IDF method, just as it is done with a

web page. We then use the rank provided by Wiki-TF-IDF to select keywords from this text, associating these keywords with the page P.

3.3 Wiki-Categories-2

The Wiki-Categories-2 method works like the Wiki-Categories-1 method, with the following difference. In step 3, to rank these categories (using the label of the categories) we use the weight of the keyword that was used to select it. For example, in Table 1, if the keyword "munition" has weight 0.8, the categories "firearms" and "artillery" will have the same weight, 0.8. The intuition behind Wiki-Categories-2 is that categories associated with keywords with a strong relation to a page P also have a strong relation to P. Notice that if the weight assigned by Wiki-TF-IDF to a keyword in a page is high, then its relation to the page is strong.

Table 1: Example of categories related to three Wikipedia titles.

Title	Categories
munition	firearms
	artillery
caliber	arms
	artillery
soldier	military forces
	military ranks

4. EXPERIMENTAL EVALUATION

In this section we analyze the performance of our proposed methods as part of a content targeted advertising system. Our keyword extraction methods were responsible for extracting a set of keywords from the target web page where the ads are presented. These keywords were then submitted as a query to a search system in order to select and rank advertising stored in an ad inventory database.

4.1 Datasets

A problem we endured in the experiments is that real data collections of ads are not publicly available for experiments. We then decided to create a dataset composed of 3,016,544 product offers extracted from an online shopping company called Nhemu[1]. The Nhemu is a price comparison service that crawls product offers from a large set of brazilian e-commerce shoppings. In this paper we considered that each product offer is described through the concatenation of three distinct attributes: name, brand, and category. This collection is referred to as *Ad-Collection-1* from now on.

While we recognize that this collection is different from the ones available on ad networks, such as the ones maintained by Google and Yahoo, it has the advantage of being now public, which will allow easy comparison of our results with future work. Further, we believe it will also be useful for future research in the area of content targeted advertising. The list of products available in the dataset is quite extensive, including books, CDs, electronic products, furniture, car accessories, games, groceries, clothes, shoes, and almost every type of product sold on the Internet, thus being a quite rich sample of product offers. Further, the books included in the collection cover all themes usually found in a library, thus opening possibilities to matches with pages about almost every topic. Given these properties, we believe that it can be used as a good reference collection to compare the effectiveness of keyword selection strategies for advertising. Further, the announcement of products constitutes a quite common type of advertising usually shown on the Internet.

We also used a real advertising collection composed by 93,972 ads from 1,744 distinct advertisers, which is referred to as *Ad-Collection-2* from now on.

As target pages (i.e.: the pages where the ads should be displayed) we used the following two sets of pages:

P_{Wiki}: a set of 300 random web pages obtained from the portuguese version of Wikipedia[2].

P_{News}: a set of 300 pages extracted from a brazilian newspaper[3].

As we have no preference for particular topics, both sets of pages cover diverse subjects, such as culture, music, personalities, sports, politics, technologies among others. As we use the Wikipedia database to enhance the representation of pages, our method is particularly good to extract keywords from pages that belong to Wikipedia.

We combined the ads and target pages into three distinct scenarios where the experiments were conducted, as presented in Table 2.

	Scenario 1	**Scenario 2**	**Scenario 3**
Pages	P_{Wiki}	P_{News}	P_{News}
Ads	Ad-Collection-1	Ad-Collection-1	Ad-Collection-2

Table 2: *Three scenarios where the experiments were conducted.*

The Wikipedia database used to compute the IDF and the co-occurrence of terms was a dump downloaded on February 2010, from which we obtained 533,358 distinct titles, including full articles, stub articles, disambiguation pages, category pages, list pages, and redirections. The product offer dataset, the web pages adopted in the experiments, and the relevance judgment of the ads associated with each web page will be available for future experiments. The reader can directly contact the authors to obtain the collection.

4.2 Evaluation Methodology

We evaluated the quality of the ads retrieved using the set of keywords extracted from target pages by each method. Each set of keywords was used as a query submitted to a retrieval system which returned a ranking of ads based on each set. The relevance judgment was performed by a group of 30 volunteers, each evaluating the ads returned by the methods considering an average of 10 web target pages.

Volunteers were asked to evaluate each retrieved ad as "relevant" or "non-relevant" in relation to a source web page. They were oriented to consider an ad as relevant if they considered that a user who was reading the page would likely click the ad presented. Given a web page and an ad collection, we presented to the users the union of results provided by all variants of the methods studied. The results for a

[1]http://nhemu.com

[2]http://pt.wikipedia.org

[3]http://www.folha.uol.com.br/

page were presented in a random order to avoid a possible bias caused by the order of results.

The system adopted to process the queries and selected ads is the Lucene[4], configured to rank documents using the vector space model. The keywords composed of more than one word, such as "South Africa", were submitted as phrases to Lucene for sake of simplicity. In a practical advertising system, a better option would be to change the indexing system to detect these keywords when indexing the ad collection, thus allowing fast search for keywords composed of more than one word.

The methods "TF-IDF", "Keyphraseness", "Wiki-TF-IDF", "Wiki-Categories-1" and "Wiki-Categories-2" provide a ranking of keywords and associate a weight with each of these keywords. We used as a query the top n keywords of each method ($1 \leq n \leq 30$) and included the computed weight for each keyword as part of the query (Lucene allows the assignment of weights to the words in its query processing interface). As "All terms" returns a set of unordered terms we used as keywords all the terms from the set. Thus, their results are presented as an horizontal straight line on the graphics. The "All terms" has an average of 185 keywords per web page. For the methods "TF-IDF", "Keyphraseness", "Wiki-TF-IDF", "Wiki-Categories-1" and "Wiki-Categories-2" we submitted the keywords with their weights, instructing the query processor to take the weights into account in the ranking.

The search results were evaluated using precision at 3 (p@3), which expresses the average percentage of relevant ads in the top 3 answers provided by the search system, as follows:

$$p@3 = \frac{|rel \cap answers|}{|answers|} \quad (2)$$

where rel is the set of relevant ads associated with the web page in the pool, $answers$ is the set of the top three ads displayed to this page by the evaluated method. Note that, as we consider only the top 3 results for each page, the maximum value of $|answers|$ is 3. Indeed, in some cases, the system retrieves less then 3 ads for the set of keywords used as query. The $p@3$ of a method is the average $p@3$ considering only the pages where at least one ad was returned.

A problem found when computing P@3 is that it is common to find cases where a method does not provide answers. This problem in fact might be quite common even in real case advertising collections, since ad collections may not cover the whole set of keywords and topics found on the web. In these cases, the precision for the specific query cannot be determined and we removed these cases when computing the final average value of P@3. Other option could be set the P@3 to 1 or 0, but in both cases the result would not reflect the reality.

Thus, to provide more insight about the results, we also introduced the computation of $recall@3$, which was calculated taking the number of relevant ads found in the pool as the set of relevant answers, but limiting this number to the maximum of relevant ads that could be shown by each method. Thus, the $recall@3$ of a method given a page p is:

$$recall@3 = \frac{|rel \cap answers|}{min(|rel|, 3)} \quad (3)$$

where rel and $answers$ are defined as in the previous equation and min is a function that returns the minimum value of two arguments.

[4]http://lucene.apache.org

4.3 Baselines

We used three different methods to compare the effectiveness of our approach, as follows:

1. "All terms": We simply used all the terms (except stopwords) in the web page to retrieve items, i.e., we used no keyword extraction algorithm at all.

2. We used a plain TF-IDF method [13] that uses words as semantic units. We used Wikipedia to calculate the Inverse Document Frequencies. We adopted this strategy because TF-IDF is a method largely used to select keywords from a text. Further, using the TF-IDF extracted from Wikipedia (the same source from where we select keywords using our proposed method) we are able to evaluate the improvements of our method when compared to this previously used strategy. Note that as we use TF-IDF information extracted from Wikipedia, TF-IDF can also be considered as a variant of TF-IDF proposed by us to take advantage of Wikipedia information in order to select keywords from web pages. However, we include it as a baseline, since our contribution in this case is marginal.

3. We used for comparison purposes a method called *Keyphraseness* [12], which was previously used as part of a successful strategy to extract keywords from web pages using information extracted from Wikipedia [5]. This method uses the probability of a term t_i to be selected as a keyword in a new document. It considers as keywords the terms that are linked to other Wikipedia articles, i.e., it considers link identification and keyword extraction as the same problem. The *Keyphraseness* of a term t_i is given by:

$$Keyphraseness(t_i) = P(is\ link|t_i) \approx \frac{n_{link}}{n_i} \quad (4)$$

where n_{link} is the number of documents the term t_i occurs as a link and n_i is the number of documents where the term occurs at least once. The extracted keywords are ranked and the top k keywords are used to represent the document. Following the procedure in [12], we only considered as candidate keywords the terms that occurred in more than 5 Wikipedia articles.

4.4 Results and Discussion

In this section we present the results provided by each method considering the three scenarios presented in Table 2.

4.4.1 Scenario 1

In this scenario (Wikipedia pages as target and Nhemu product offer dataset as the ad collection), the proposed methods achieved the best results. This happens because the target pages are also from Wikipedia, and thus naturally have content associated with the categories extracted by our method. We emphasize that providing effective mechanisms to select ads for Wikipedia web pages is itself an important application that could be used in practice.

Figure 1 presents the P@3 results obtained by our method in this scenario. First, when selecting from 1 to 20 keywords, our three methods outperform the baselines TF-IDF and Keyphraseness in terms of precision. Figure 1 also shows

that the keyword expansion methods Wiki-categories-1 and Wiki-categories-2 caused improvements in the quality of results, achieving a gain of up to 26% when compared to "All terms" and a gain of 33% when compared to TF-IDF, the second best baseline.

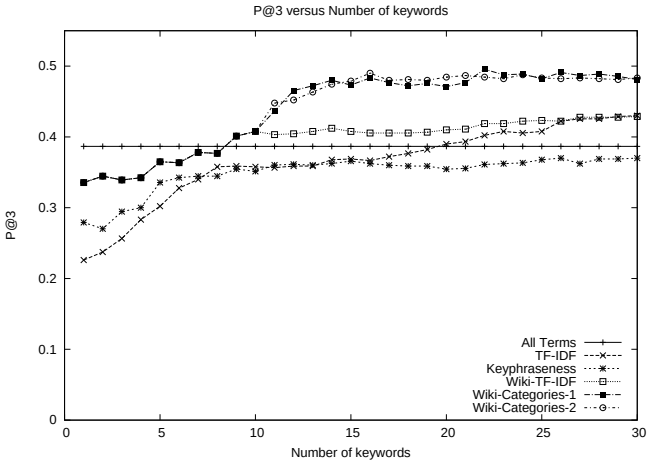

Figure 1: P@3 for each method using different number of keywords in scenario 1 (Wikipedia pages as target and Nhemu product offer dataset as the ad collection). The method "All terms" has an average of 185 keywords per web page.

The gains are statistically significant according to $t-test$ when comparing Wiki-categories-1 and Wiki-categories-2 to the baselines Keyphraseness and TF-IDF in all levels, being also significant when comparing to "All terms" and using at least 12 terms. The gains achieved by Wiki-TF-IDF are significant when compared to Keyphraseness in all points of the curve. The gains achieved by Wiki-TF-IDF are significant only for 1 to 14 keywords, being not significant when using more keywords.

An important issue in ad selection systems is the final computational costs required to select ads. While "All terms" requires an average number of 185 keywords to represent a web page, our method Wiki-TF-IDF required only about 8 keywords to achieve the same performance. This is an important property, since the computational cost to select ads is linear in the number of keywords. Thus, the expected computational cost to select ads using the keyword sets provided by our methods would be less than 1% of the cost for selecting ads when using "All terms". In this first scenario, where the target pages are extracted from Wikipedia, the improvement in time performance is obtained also with a significant improvement in the quality of the selected ads.

Figure 2 depicts the R@3 results achieved with the methods in the first scenario. As it can be seen, recall results obtained by the methods are equal to the P@3 values. This similarity in the results indicates that the ad selector has retrieved ads for virtually all the sets of keywords selected by the compared methods in scenario 1. This abundance of results is related to the large variety of products found in the collection Nhemu, which includes, for instance, a large collection of books that also covers a variety of topics.

4.4.2 Scenario 2

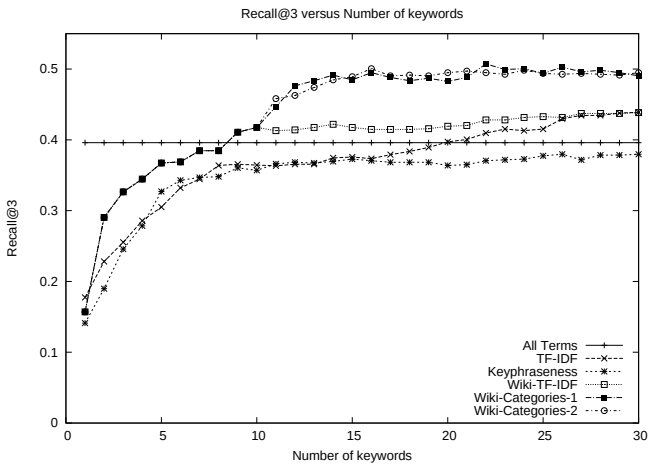

Figure 2: R@3 for each method using different number of keywords in scenario 1 (Wikipedia pages as target and Nhemu product offer dataset as the ad collection). The method "All terms" have a average of 185 keywords per web page.

Figure 3 presents the P@3 results obtained by our method in Scenario 2. In this second scenario, we keep Nhemu dataset as the ad collection, but changed the target pages to news pages. This change in the target pages is important to avoid the bias in favor of category methods achieved when using Wikipedia pages as targets.

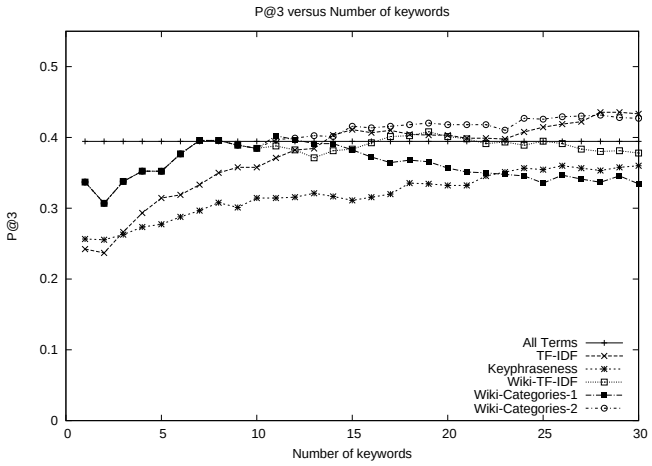

Figure 3: P@3 for each method using different number of keywords in scenario 2 (News pages as target and Nhemu product offer dataset as the ad collection). The method "All terms" have a average of 185 keywords per web page.

When selecting from 1 to 8 keywords, method Wiki-TF-IDF again outperforms the baselines TF-IDF and Keyphraseness in terms of precision, with gains statistically significant. Figure 3 also shows the keyword expansion method Wiki-Categories-2 caused just a low improvement in the quality of results, with gains achieved being not significant when compared to Wiki-TF-IDF. Method Wiki-Categories-2 in fact has caused loss of quality in this scenario.

When checking the R@3 values presented in Figure 4, conclusions are close to the ones obtained in scenario 1. The usage of Nhemu collection again allowed the values of precision and recall to be almost the same, expect for very small number of keywords (less than 3).

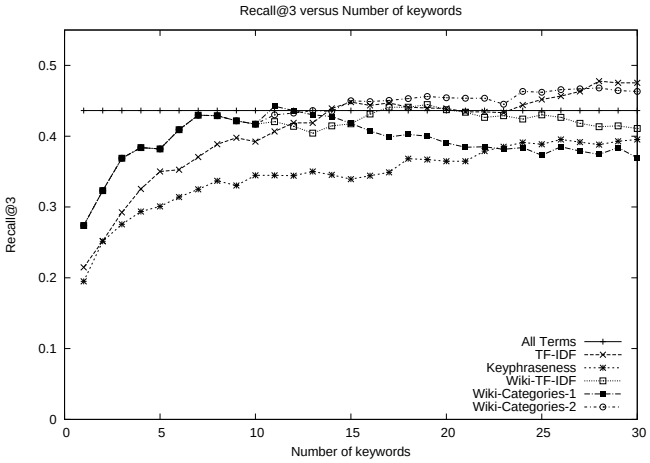

Figure 4: R@3 for each method using different number of keywords in scenario 2 (News pages as target and Nhemu product offer dataset as the ad collection). The method "All terms" have a average of 185 keywords per web page.

4.4.3 Scenario 3

In scenario 3, we compare the methods when the target pages contain news and the ads are extracted from a real case ad collection. Figure 5 depicts the results. Note that again method Wiki-TF-IDF achieved gains when compared to Keyphraseness. However, we notice that in this scenario the differences between our method and TF-IDF are not significant. Further, all the experimented methods are worse than an approach using all terms, with differences being statistically significant for all cases where the methods select less than 10 keywords.

Figure 5 also shows that method Wiki-Categories-2 is slightly better than Wiki-Categories-1, while again the category expansion resulted in no significant improvement in the results when compared to Wiki-TF-IDF.

Figure 6 presents the R@3 curves in this third scenario. The recall in this case is worse than precision when using just a few keywords, from 1 to 6 keywords. That means this scenario presents more cases where the ad selector was not able to find any match between the given keywords and the ad collection. This result was expected, since in this scenario the ad collection contains fewer and less diverse elements than the Nhemu database adopted in scenarios 1 and 2. Further, we can see that this scenario is less favorable to our methods. On the other hand, even in this less favorable scenario still our methods are quite competitive when compared to the baselines.

5. CONCLUSIONS AND FUTURE WORK

In this work we proposed three novel approaches for selecting keywords on Web pages: Wiki-TF-IDF, Wiki-Categories-1 and Wiki-Categories-2. The objective was to reduce noise and improve the matching between the keywords and ads in an advertising system. To evaluate the proposed methods we used the software Lucene as the ad selector.

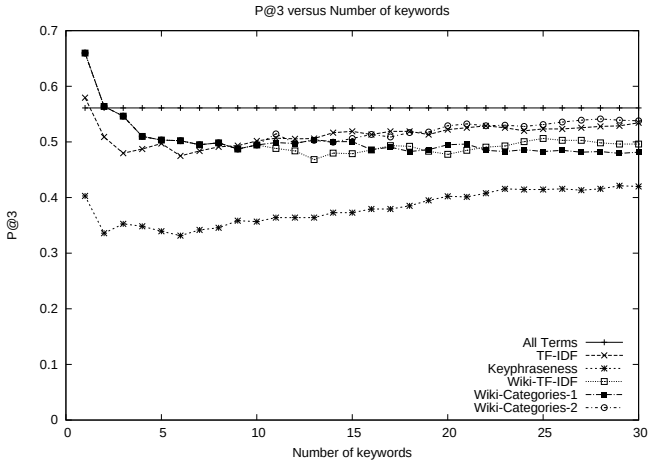

Figure 5: P@3 for each method using different number of keywords in scenario 3 (News pages as target and the ad collection dataset). The method "All terms" have a average of 185 keywords per web page.

Experimental results have shown that the three methods are competitive in practice. For instance, when selecting keywords from Wikipedia articles, our best method outperformed the representations based on all the terms (i) and TF-IDF weighting (iii) with gains of about 33% and 26 %, respectively. In the worst scenario we found in our experiments, our methods achieved results similar to the approach proposed by [12].

We also have shown that Wiki-Categories-1 and Wiki-Categories-2 presented a good performance using small sets of keywords to represent each web page, presenting small computational costs. They achieved, in some scenarios, results with a quality even superior to the other methods experimented. For instance, on Scenario 1 they achieved results slightly superior to the ones obtained when using all terms of a page, which could be a trivial solution to the problem of representing a web page.

Although we experimented the proposed keyword selection methods only with advertising systems, the results presented indicate they may be specially useful in any application where there is a requirement of representing the content of a web page with an small set of keywords. This is the case, for instance, when this small set of keywords is used as a query to an API of web service that limits the maximum number of keywords in a query. For instance, the methods proposed could be useful to automatically select videos related to a page from online video servers, such as Youtube, or to automatically recommend books when a user is browsing a page in a Web site. We will further investigate these and other applications to our method as future work. Finally, since our methods are very cheap to compute, they could be used as complementary features for more sophisticated strategies such as the ones based in machine learning, described in Section 2.

6. REFERENCES

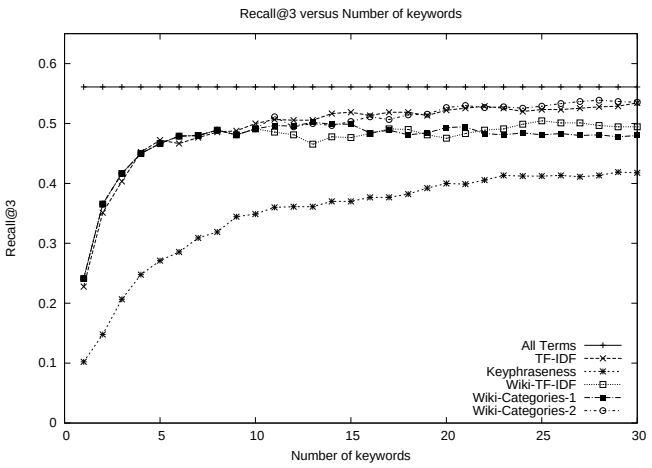

Figure 6: R@3 for each method using different number of keywords in scenario 3 (News pages as target and the ad collection dataset). The method "All terms" have a average of 185 keywords per web page.

[1] R. A. Baeza-Yates and B. A. Ribeiro-Neto. *Modern Information Retrieval*. Pearson / Addison-Wesley, second edition, 2011.

[2] D. Carmel, H. Roitman, and N. Zwerdling. Enhancing cluster labeling using wikipedia. In *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 139–146, 2009.

[3] E. Gabrilovich and S. Markovitch. Wikipedia-based semantic interpretation for natural language processing. *J. Artif. Int. Res.*, 34:443–498, March 2009.

[4] J. Goodman and V. R. Carvalho. Implicit queries for email. In *Second Conference on Email and Anti-Spam*, `http://www.ceas.cc/papers-2005/141.pdf`, 2005.

[5] M. Grineva, M. Grinev, and D. Lizorkin. Extracting key terms from noisy and multitheme documents. In *Proceedings of the 18th International Conference on the World wide web*, pages 661–670, 2009.

[6] J. Hu, L. Fang, Y. Cao, H.-J. Zeng, H. Li, Q. Yang, and Z. Chen. Enhancing text clustering by leveraging wikipedia semantics. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 179–186, 2008.

[7] X. Hu, X. Zhang, C. Lu, E. K. Park, and X. Zhou. Exploiting wikipedia as external knowledge for document clustering. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 389–396, 2009.

[8] A. Huang, D. Milne, E. Frank, and I. H. Witten. Clustering documents with active learning using wikipedia. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, pages 839–844, 2008.

[9] A. Huang, D. Milne, E. Frank, and I. H. Witten. Clustering documents using a wikipedia-based concept representation. In *Proceedings of the 13th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*, pages 628–636, 2009.

[10] U. Irmak, V. von Brzeski, and R. Kraft. Contextual ranking of keywords using click data. In *Proceedings of the 2009 IEEE International Conference on Data Engineering*, pages 457–468, 2009.

[11] M. Koolen, G. Kazai, and N. Craswell. Wikipedia pages as entry points for book search. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 44–53, 2009.

[12] R. Mihalcea and A. Csomai. Wikify!: linking documents to encyclopedic knowledge. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, pages 233–242, 2007.

[13] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage.*, 24(5):513–523, 1988.

[14] W.-t. Yih, J. Goodman, and V. R. Carvalho. Finding advertising keywords on web pages. In *Proceedings of the 15th International Conference on the World Wide Web*, pages 213–222, 2006.

[15] B. Zhou, P. Luo, Y. Xiong, and W. Liu. Wikipedia-graph based key concept extraction towards news analysis. In *Proceedings of the 2009 IEEE Conference on Commerce and Enterprise Computing*, pages 121–128, 2009.