Experimenting Text Summarization Techniques for Contextual Advertising

Giuliano Armano, Alessandro Giulian, and Eloisa Vargiu

University of Cagliari
Department of Electrical and Electronic Engineering
{armano, alessandro.giuliani, vargiu}@diee.unica.it
http://iasc.diee.unica.it

Abstract. Contextual advertising systems suggest suitable advertisings to users while surfing the Web. Focusing on text summarization, we propose novel techniques for contextual advertising. Comparative experiments between these techniques and existing ones have been performed.

Keywords: contextual advertising, information retrieval and filtering

1 Introduction

Most of the advertisements on the Web are short textual messages, usually marked as "sponsored links". Two main kinds of textual advertising approaches are used on the Web today [8]: sponsored search and contextual advertising. The former puts advertisements (ads) on the pages returned from a Web search engine following a query. All major current Web search engines support this kind of ads, acting simultaneously as search engine and advertisement agency. The latter puts ads within the content of a generic, third party, Web page. A commercial intermediary, namely an ad-network, is usually in charge of optimizing the selection of ads. In other words, contextual advertising (CA hereinafter) is a form of targeted advertising for ads appearing on websites or other media, such as contents displayed in mobile browsers. Ads are selected and served by automated systems based on the content displayed to the user.

We consider a scenario of online advertising, in which an intermediating commercial net (ad-network) is responsible for optimizing the selection of ads. The goal is twofold: (i) increasing commercial company revenues and (ii) improving user experience. Let us point out in advance that, in information retrieval, the term "context" may have different interpretations depending on the research field. For instance, it denotes "event which modify the user behavior in the field of recommender systems". For CA it denotes "keywords used in search engines".

A CA system typically involves four main tasks: (i) pre-processing, (ii) text summarization, (iii) classification, and (iv) matching. In this paper, we are mainly interested in text summarization, which is aimed at generating a short representation of a textual document (e.g., a Web page) with negligible loss of information.

Starting from state-of-the-art text-summarization techniques, we propose new and more effective techniques. Then, we perform comparative experiments to assess the effectiveness of the proposed techniques. Preliminary results show that the proposed techniques perform better than existing ones.

The paper is organized as follows. First, the main work on CA is briefly recalled. Subsequently, text summarization is illustrated from both a generic perspective and in the context of CA. After illustrating an implementation of a CA system, preliminary experimental results are then reported and discussed. Conclusions and future directions end the paper.

2 Contextual Advertising

As discussed in [6], CA is an interplay of four players:

- The *advertiser* provides the supply of ads. Usually the activity of the advertisers is organized around campaigns which are defined by a set of ads with a particular temporal and thematic goal (e.g., sale of digital cameras during the holiday season). As in traditional advertising, the goal of the advertisers can be broadly defined as the promotion of products or services.
- The *publisher* is the owner of the Web pages on which the advertising is displayed. The publisher typically aims to maximize advertising revenue while providing a good user experience.
- The *ad network* is a mediator between the advertiser and the publisher; it selects the ads to display on the Web pages. The ad-network shares the advertisement revenue with the publisher.
- The *Users* visit the Web pages of the publisher and interact with the ads.

Ribeiro-Neto et al. [22] examine a number of strategies to match pages and ads based on extracted keywords. Ads and pages are represented as vectors in a vector space. To deal with semantic problems that may arise from a pure keyword-based approach, the authors expand the page vocabulary with terms from similar pages weighted according to their similarity to the matched page. In a subsequent work, the authors propose a method to learn the impact of individual features using genetic programming [16].

Another approach to CA is to reduce it to the problem of sponsored search by extracting phrases from a Web page and matching them with the bid phrases of each ad. In [26], a system for phrase extraction is proposed, which uses a variety of features to determine the importance of page phrases for advertising purposes. The system is trained with pages that have been annotated by hand with important phrases. In [6], the same approach is used, with a phrase extractor based on the work reported in [25].

3 Text Summarization

Radev et al. [21] define a summary as "a text that is produced from one or more texts, that conveys important information in the original text(s), and that

is no longer than half of the original text(s) and usually significantly less than that". This simple definition highlights three important aspects that characterize research on automatic summarization: (i) summaries may be produced from a single document or multiple documents; (ii) summaries should preserve important information; and (iii) summaries should be short. Unfortunately, attempts to provide a more elaborate definition for this task resulted in disagreement within the community [7].

Summarization techniques can be divided into two groups [15]: (i) those that extract information from the source documents (extraction-based approaches) and (ii) those that abstract from the source documents (abstraction-based approaches). The former impose the constraint that a summary uses only components extracted from the source document, whereas the latter relax the constraints on how the summary is created. Extraction-based approaches are mainly concerned with what the summary content should be, usually relying solely on extraction of sentences. On the other hand, abstraction-based approaches put strong emphasis on the form, aiming to produce a grammatical summary, which usually requires advanced language generation techniques. Although potentially more powerful, abstraction-based approaches have been far less popular than their extraction-based counterparts, mainly because generating the latter is easier. In a paradigm more tuned to information retrieval, one can also consider topic-driven summarization, which assumes that the summary content depends on the preference of the user and can be assessed via a query, making the final summary focused on a particular topic. In this paper, we exclusively focus on extraction-based methods.

An extraction-based summary consists of a subset of words from the original document and its bag of words representation can be created by selectively removing a number of features from the original term set. In text categorization, such process is known as feature selection and is guided by the "usefulness" of individual features as far as the classification accuracy is concerned. However, in the context of text summarization, feature selection is only a secondary aspect. It might be argued that in some cases a summary may contain the same set of features as the original; for example, when it is created by removing the redundant/repetitive words or phrases. Typically, an extraction-based summary whose length is only 10-15% of the original is likely to lead to a significant feature reduction as well.

Many studies suggest that even simple summaries are quite effective in carrying over the relevant information about a document. From the text categorization perspective, their advantage over specialized feature selection methods lies in their reliance on a single document only (the one that is being summarized) without computing the statistics for all documents sharing the same category label, or even for all documents in a collection. Moreover, various forms of summaries become ubiquitous on the Web and in certain cases their accessibility may grow faster than that of full documents.

Earliest instances of research on summarizing scientific documents proposed paradigms for extracting salient sentences from text using features like word and

phrase frequency [17], position in the text [3], and key phrases [10]. Various works published since then had concentrated on other domains, mostly on newswire data. Many approaches addressed the problem by building systems dependent on the type of the required summary.

Simple summarization-like techniques have been long applied to enrich the set of features used in text categorization. For example, a common strategy is to give extra weight to words appearing in the title of a story [19] or to treat the title-words as separate features, even if the same words were present elsewhere in the text body [9]. It has been also noticed that many documents contain useful formatting information, loosely defined as context, that can be utilized when selecting the salient words, phrases or sentences. For example, Web search engines select terms differently according to their HTML markup [4]. Summaries, rather than full documents, have been successfully applied to document clustering [11]. Ker and Chen [13] evaluated the performance of a categorization system using title-based summaries as document descriptors. In their experiments with a probabilistic TF-IDF based classifier, they shown that title-based document descriptors positively affected the performance of categorization.

4 Text Summarization in Contextual Advertising

As the input of a contextual advertiser is an HTML document, contextual advertising systems typically rely on extraction-based approaches, which are applied to the relevant blocks of a Web page (e.g., the title of the Web page, its first paragraph, and the paragraph which has the highest title-word count).

In the work of Kolcz et al. [15] seven straightforward (but effective) extraction-based text summarization techniques have been proposed and compared. In all cases, a word occurring at least three times in the body of a document is a keyword, while a word occurring at least once in the title of a document is a title-word. For the sake of completeness, let us recall the proposed techniques:

- *Title* (T), the title of a document;
- *First Paragraph* (FP), the first paragraph of a document;
- *First Two Paragraphs* (F2P), the first two paragraphs of a document;
- *First and Last Paragraphs* (FLP), the first and the last paragraphs of a document;
- *Paragraph with most keywords* (MK), the paragraph that has the highest number of keywords;
- *Paragraph with most title-words* (MT), the paragraph that has the highest number of title-words;
- *Best Sentence* (BS), sentences in the document that contain at least 3 title-words and at least 4 keywords.

One may argue that the above methods are too simple. However, as shown in [5], extraction-based summaries of news articles can be more informative than those resulting from more complex approaches. Also, headline-based article descriptors proved to be effective in determining user's interests [14].

Our proposal consists of enriching some of the techniques introduced by Kolcz et al. with information extracted from the title, as follows:

- *Title and First Paragraph* (TFP), the title of a document and its first paragraph:
- *Title and First Two Paragraphs* (TF2P), the title of a document and its first two paragraphs;
- *Title, First and Last Paragraphs* (TFLP), the title of a document and its first and last paragraphs;
- *Most Title-words and Keywords* (MTK), the paragraph with the highest number of title-words and that with the highest number of keywords.

We also defined a further technique, called *NKeywords* (NK), that selects the N most frequent keywords.[1]

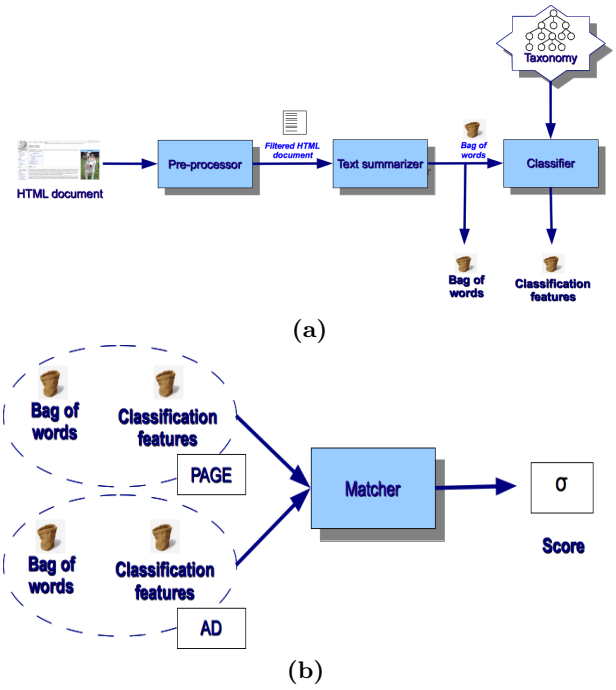

Fig. 1. A generic CA architecture at a glance.

[1] N is a global parameter that can be set starting from some relevant characteristics of the input (e.g., from the average document length).

5 The Implemented System

Our view of CA is sketched in Figure 1, which illustrates a generic architecture that can give rise to specific systems depending on the choices made on each involved module. Notably, most of the state-of-the-art solutions are compliant with this view. So far, we implemented in Java the sub-system depicted in Figure 1.a, which encompasses (i) a pre-processor, (ii) a text summarizer, and (iii) a classifier.

Pre-processor. Its main purpose is to transform an HTML document (a Web page or an ad) into an easy-to-process document in plain-text format, while maintaining important information. This is obtained by preserving the blocks of the original HTML document, while removing HTML tags and stop-words.[2] First, any given HTML page is parsed to identify and remove noisy elements, such as tags, comments and other non-textual items. Then, stop-words are removed from each textual excerpt. Finally, the document is tokenized and each term stemmed using the well-known Porter's algorithm [20].

Text summarizer. The text summarizer outputs a vector representation of the original HTML document as bag of words (BoW), each word being weighted by TF-IDF [23]. So far, we implemented the methods of Kolcz et al. (see Section 4), but not "Title" and "Best Sentence". These two methods were defined to extract summaries from textual documents such as articles, scientific papers and books. In fact, we are interested in summarizing HTML documents, in which the title is often not representative. Moreover, they are often too short to find meaningful sentences composed by at least 3 title-words and 4 keywords in the same sentence.

Classifier. Text summarization is a purely syntactic analysis and the corresponding Web-page classification is usually inaccurate. To alleviate possible harmful effects of summarization, both page excerpts and advertisings are classified according to a given set of categories [2]. The corresponding classification-based features (CF) are then used in conjunction with the original BoW. In the current implementation, we adopt a centroid-based classification technique [12], which represents each class with its centroid calculated starting from the training set. A page is classified measuring the distance between its vector and the centroid vector of each class by adopting the cosine similarity.

Matcher. It is devoted to suggest ads (a) to the Web page (p) according to a similarity score based on both BoW and CF [2]. In formula (α is a global parameter that permits to control the emphasis of the syntactic component with respect to the semantic one):

$$score(p, a) = \alpha \cdot sim_{BoW}(p, a) + (1 - \alpha) \cdot sim_{CF}(p, a) \quad (1)$$

[2] To this end, the Jericho API for Java has been adopted, described at the Web page: http://jericho.htmlparser.net/docs/index.html

where, $sim_{BoW}(p, a)$ and $sim_{CF}(p, a)$ are cosine similarity scores between p and a using BoW and CF, respectively. This module has not been implemented yet. However, it is worth recalling that in this paper we are interested in making comparisons among text summarization techniques.

6 Preliminary Results

We performed experiments aimed at comparing the techniques described in Section 4. To assess them we used the BankSearch Dataset [24], built using the Open Directory Project and Yahoo! Categories[3], consisting of about 11000 Web pages classified by hand in 11 different classes.

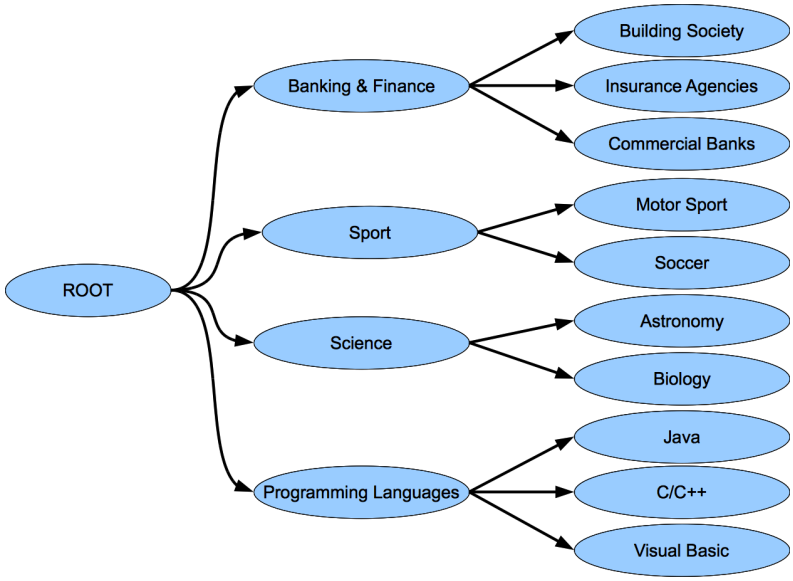

Fig. 2. Class hierarchy of BankSearch Dataset.

Figure 2 shows the overall hierarchy. The 11 selected classes are the leaves of the taxonomy, together with the class *Sport*, which contains web documents from all the sites that were classified as sport, except for the sites that were classified as *Soccer* or *Motor Sport*. In [24], the authors show that this structure provides a good test not only for generic classification/clustering methods, but also for hierarchical techniques.

Table 1 shows the performances in terms of accuracy (A), macro-precision (P), and macro-recall (R). For each technique, the average number of unique

[3] http://www.dmoz.org and http://www.yahoo.com, respectively

Table 1. Results of text summarization techniques comparison.

	FP	**F2P**	**FLP**	**MK**	**MT**	**TFP**	**TF2P**	**TFLP**	**MTK**	**NK**
A	0.598	0.694	0.743	0.608	0.581	0.802	0.821	0.833	0.721	0.715
P	0.606	0.699	0.745	0.702	0.717	0.802	0.822	0.832	0.766	0.722
R	0.581	0.673	0.719	0.587	0.568	0.772	0.789	0.801	0.699	0.693
T	13	24	24	25	15	16	27	26	34	10

extracted terms (T) is shown. For NKeywords summarization, we performed experiments with N=10.

As a final remark, let us note that just adding information about the title improves the performances of summarization. Another interesting result is that, as expected, the TFLP summarization provides the best performance, as FLP summarization does for the classic techniques.

7 Conclusions and Future Directions

In this paper, we presented a preliminary study on text summarization techniques applied to CA. In particular, we proposed some straightforward extraction-based techniques that improve those proposed in the literature. Experimental results confirm the hypothesis that adding information about titles to well-known techniques allows to improve performances.

As for future directions, we are currently studying a novel semantic technique. The main idea is to improve syntactic techniques by exploiting semantic information (such as, synonyms and hypernyms) extracted from a lexical database (e.g., WordNet [18]) in conjunction with a POS-tagging and word sense disambiguation. Further experiments are also under way. In particular, we are setting up the system to calculate its performances with a larger dataset extracted by DMOZ in which documents are categorized according to a given taxonomy of classes. Moreover, as we deem that bringing ideas from recommender systems will help in devising CA systems [1], we are also studying a collaborative approach to CA.

Acknowledgments. This work has been partially supported by Hoplo srl. We wish to thank, in particular, Ferdinando Licheri and Roberto Murgia for their help and useful suggestions.

References

1. A. Addis, G. Armano, A. Giuliani, and E. Vargiu. A recommender system based on a generic contextual advertising approach. In *Proceedings of ISCC'10: IEEE Symposium on Computers and Communications*, pages 859–861, 2010.
2. A. Anagnostopoulos, A. Z. Broder, E. Gabrilovich, V. Josifovski, and L. Riedel. Just-in-time contextual advertising. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 331–340, New York, NY, USA, 2007. ACM.

3. P. Baxendale. Machine-made index for technical literature - an experiment. *IBM Journal of Research and Development*, 2:354–361, 1958.
4. R. K. Belew. *Finding out about: A Cognitive Perspective on Search Engine Technology and the WWW*. Cambridge University Press, 2000.
5. R. Brandow, K. Mitze, and L. F. Rau. Automatic condensation of electronic publications by sentence selection. *Inf. Process. Manage.*, 31:675–685, September 1995.
6. A. Broder, M. Fontoura, V. Josifovski, and L. Riedel. A semantic approach to contextual advertising. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 559–566, New York, NY, USA, 2007. ACM.
7. D. Das and A. F. Martins. A survey on automatic text summarization. Technical Report Literature Survey for the Language and Statistics II course at CMU, 2007.
8. C. Deepayan, A. Deepak, and J. Vanja. Contextual advertising by combining relevance with click feedback. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 417–426, New York, NY, USA, 2008. ACM.
9. S. Dumais, J. Platt, D. Heckerman, and M. Sahami. Inductive learning algorithms and representations for text categorization. In *Proceedings of the seventh international conference on Information and knowledge management*, CIKM '98, pages 148–155, New York, NY, USA, 1998. ACM.
10. H. P. Edmundson. New methods in automatic extracting. *Journal of ACM*, 16:264–285, April 1969.
11. V. Ganti, J. Gehrke, and R. Ramakrishnan. Cactus: clustering categorical data using summaries. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '99, pages 73–83, New York, NY, USA, 1999. ACM.
12. E.-H. Han and G. Karypis. Centroid-based document classification: Analysis and experimental results. In *Proceedings of the 4th European Conference on Principles of Data Mining and Knowledge Discovery*, PKDD '00, pages 424–431, London, UK, 2000. Springer-Verlag.
13. S. J. Ker and J.-N. Chen. A text categorization based on summarization technique. In *Proceedings of the ACL-2000 workshop on Recent advances in natural language processing and information retrieval: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics - Volume 11*, pages 79–83, Morristown, NJ, USA, 2000. Association for Computational Linguistics.
14. A. Koĺcz and J. Alspector. Asymmetric missing-data problems: Overcoming the lack of negative data in preference ranking. *Inf. Retr.*, 5:5–40, January 2002.
15. A. Kolcz, V. Prabakarmurthi, and J. Kalita. Summarization as feature selection for text categorization. In *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*, pages 365–370, New York, NY, USA, 2001. ACM.
16. A. Lacerda, M. Cristo, M. A. Gonçalves, W. Fan, N. Ziviani, and B. Ribeiro-Neto. Learning to advertise. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 549–556, New York, NY, USA, 2006. ACM.
17. H. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2:159–165, 1958.
18. G. A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, 1995.

19. D. Mladenić and M. Grobelnik. Feature selection for classification based on text hierarchy. In *Text and the Web, Conference on Automated Learning and Discovery CONALD-98*, 1998.
20. M. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.
21. D. R. Radev, E. Hovy, and K. McKeown. Introduction to the special issue on summarization. *Computational Linguistic*, 28:399–408, December 2002.
22. B. Ribeiro-Neto, M. Cristo, P. B. Golgher, and E. Silva de Moura. Impedance coupling in content-targeted advertising. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 496–503, New York, NY, USA, 2005. ACM.
23. G. Salton and M. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill Book Company, 1984.
24. M. Sinka and D. Corne. A large benchmark dataset for web document clustering. In *Soft Computing Systems: Design, Management and Applications, Volume 87 of Frontiers in Artificial Intelligence and Applications*, pages 881–890. Press, 2002.
25. R. Stata, K. Bharat, and F. Maghoul. The term vector database: fast access to indexing terms for web pages. *Comput. Netw.*, 33(1-6):247–255, 2000.
26. W.-t. Yih, J. Goodman, and V. R. Carvalho. Finding advertising keywords on web pages. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 213–222, New York, NY, USA, 2006. ACM.