Improving Text Summarization Using Noun Retrieval Techniques

Christos Bouras and Vassilis Tsogkas

Research Academic Computer Technology Institute, N. Kazantzaki, Panepistimioupoli and Computer Engineering and Informatics Department, University of Patras, Greece
{bouras,tsogkas}@cti.gr

Abstract. Text Summarization and categorization have always been two of the most demanding information retrieval tasks. Deploying a generalized, multi-functional mechanism that produces good results for both of the aforementioned tasks seems to be a panacea for most of the text-based, information retrieval needs. In this paper, we present the keyword extraction techniques, exploring the effects that part of speech tagging has on the summarization procedure of an existing system.

Keywords: Focused Crawler, Part of Speech Tagging, Noun Retrieval, Data Preprocessing, Text Summarization, Text Categorization.

1 Introduction

Keyword extraction, being the basis of any information retrieval (IR) task, aims to select the appropriate keywords out of a text, accompanying them with a suitable score that depicts their importance. With the term appropriate, we mean the most representative words, as far as the text's overall meaning is concerned. Following the keyword extraction procedure, text summarization and categorization techniques come. In our research, a unified, yet autonomous system is developed, PeRSSonal [1], in which summarization and categorization are the core procedures (as well as personalization of the presented results). This paper studies the improvement of the aforementioned procedures by assisting our keyword extraction mechanism with noun retrieval capabilities.

Presenting to the user summaries matching their needs is a very crucial procedure that can assist information filtering. Even though automatic text summarization dates back to Luhn's work in the 1950's, several researchers continued investigating various approaches to the summarization problem up to nowadays. A summary [2] usually helps readers identify interesting articles or even understand the overall story about an event. Most of the times, the summarization approaches are based on a "text-span level" [3], with sentences being the most common type of text-span having each of them rated according to some criteria (e.g. important keywords, lexical chains, etc.). These techniques transform the original problem to a simpler one: ranking sentences according to their salience or likelihood of being part of a summary, concatenating them at a second stage. Some techniques [4] try to identify special words and phrases in the text, while in [5] the authors compare patterns of relationships between the sentences.

I. Lovrek, R.J. Howlett, and L.C. Jain (Eds.): KES 2008, Part II, LNAI 5178, pp. 593–600, 2008.

Typical classification tasks are deciding to what folder an email message should be directed, on which newsgroup a news article belongs, etc. Several text classification (categorization) approaches have been researched over the years: Naive Bayesian(NB), K-Nearest Neighbor(KNN), and Centroid-based(CB) techniques are some examples. New articles can be categorized to the pre-defined categories using some criteria which vary from one technique to another. Categories can be relatively coarse-grained, i.e. only some basic unrelated to each other, or fine-grained, where many categories, frequently overlapping with each other, are introduced. Linear Least Squares (LLSF) [6], a multivariate regression model that is automatically learned from a training set of documents and their categories, gives good results and is utilized in our work.

Automatic part of speech tagging, is a well known problem that has been addressed by several researchers during the last twenty years. It is a firm belief that when it comes to keyword extraction, the nouns of the text carry most of the sentence meaning. In a sense, extracted nouns should lead to better semantic representation of the text, and hence, improved IR results. Noun extraction, a subtask of POS tagging, is the process of identifying every noun (either proper or common) in an article or a document. In many languages, nouns are used as the most important terms (features) that express a document's meaning in NLP applications such as information retrieval, document categorization, text summarization, information extraction, etc. Various methodologies have been proposed making use of linguistic [7], statistical [8], symbolic learning knowledge [9] or support vector machines [10] and can be categorized to: morphological analysis, or POS tagging based. The former methods try to generate all possible interpretations of a given phrase by implementing a morphological analyzer or a simpler method using lexical dictionaries. It may over-generate or extract inaccurate nouns due to lexical ambiguity and shows a low precision rate. On the other hand, the POS tagging based methods choose the most probable analysis among the results produced by the morphological analyzer. Due to the resolution of the ambiguities, it can obtain relatively accurate results. However, it also suffers from errors not only produced by the POS tagger, but also triggered by the preceding morphological analyzer.

In this paper we present the incorporation of noun retrieval techniques in PeRSSonal, using the SVM method for POS tagging, as part of its keyword extraction algorithms, and we explore, though experimentation, the possible improvements this change has on the mechanism's IR procedures: summarization and categorization.

In the next section the architecture of the proposed mechanism is introduced. In Section 3 the algorithm analysis of the mechanism is presented. Section 4 describes the experimental procedure that took place and its results. Section 5 concludes and outlines the directions of possible future research.

2 Architecture

PeRSSonal [1] follows a classic n-tier architectural approach. The system consists of four layers which work autonomously and collaborate through a centralized database. The web interface handles the information flow into the mechanism which is then directed to the interior subsystems. Text preprocessing techniques follow and the

results are led to the next level of analysis where core IR techniques are located. Finally the outcomes are presented to the end users though the information presentation subsystem. In the current paper, we extend the text preprocessing subsystem using noun retrieval techniques. The implemented architecture is depicted in Fig. 1 and the modified component is presented in the dashed box.

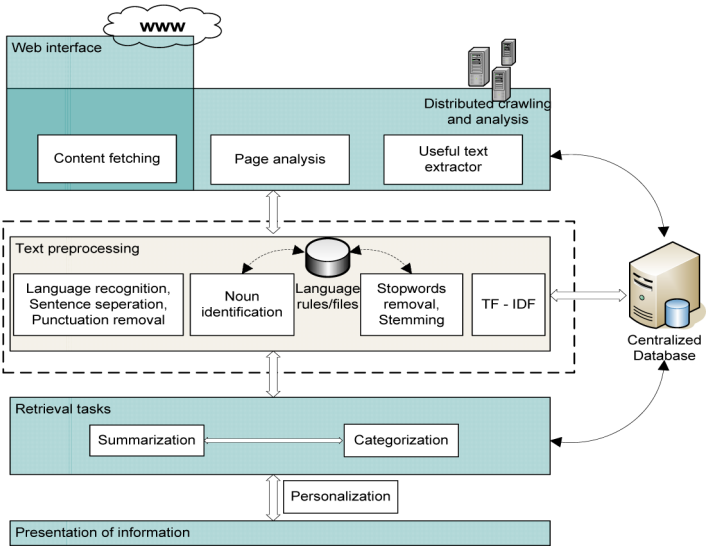

Fig. 1. System's architecture

The first layer constitutes the interconnection between the mechanism and the web sources where the following procedures take place: content fetching procedure, analysis of the downloaded content and lastly, extraction of the useful information from the web content. In order to capture web pages, a simple focused web crawler is used. The crawler takes as input the addresses that are extracted from existing RSS feeds, deriving from several major news portals. The crawling procedure is distributed across multiple systems which synchronize thought the centralized database. Crawled html pages are analyzed and are stored without any other unnecessary page element (images, css, javascript, etc.). During this analysis level, our system isolates the "useful text", meaning the main body of the article, and the database is populated with news articles that are ready for the text preprocessing step.

The second tier of the system, which is the focusing of this paper, works on the article's title and body applying several preprocessing techniques. In particular, after the retrieval of the stored article that resides in the database, a series of inner procedures take place at this layer. Firstly, the article's language is recognized either directly through language identifying procedures, or indirectly using the predetermined language of the origin-feed. Following is a sentence separation and punctuation removal step. Afterwards, the noun identification step takes place which, by utilizing the POS SVM-based tagger [10], is able to determine with high precision the article's nouns. Some common text extraction techniques follow: stopwords removal and stemming.

Noun extraction should precede these procedures if it is to succeed will high probability. It is important to note that the noun identification, stopwords removal and stemming procedures are language dependant, meaning that specific language rules, stopword lists and stemming rules respectively, have to be applied for different languages. The above set the foundations for multi-language support by our mechanism, even though only the English language has been incorporated so far. The results of the procedures described in this layer are stemmed keywords either marked as nouns or not, their location in the text and their frequency of appearance in it. These are represented through term frequency – inverse document frequency (TF-IDF) vector statistics that are stored in the database and are utilized by the procedures of the third analysis level.

The information retrieval tasks of our mechanism are located in the third analysis level, where the summarization and categorization algorithms are applied. The main scope of the categorization module is to assist the summarization procedure by pre-labeling the article with a category and has proven in [1] to be providing better results. Following the IR task of the mechanism, personalization algorithms take place and the content is finally delivered to the user.

3 Algorithm Analysis

Our analysis consists of three different algorithmic steps: extraction of keywords and identification of nouns, categorization procedure and summarization procedure.

3.1 Keyword Extraction and Noun Identification Procedure

The input to the keyword extraction module is plain text that defines the article's body and title as well as its language. Apart from the previous, some parameters have to be tuned in order for the mechanism to be the most efficient: a) minimum word length (all words with length smaller than the minimum are removed) and b) the language dependant stopword list that will be used. Our experimentation for news articles in [11] revealed that a limit of 5 letters is the best suited as far as articles written in English are concerned.

Noun identification involves an off-line learning step for the POS tagger using language specific rules. Previous to the tagging, SVM models (weight vectors and biases) are learned from a training corpus using the learning component. A modified version of the SVMTool [10] is used so that tagging takes place only for nouns, saving system's processing time. Once the training is complete, the article's body is forwarded to the tagger and the text's nouns are marked. Stopwords removal takes place and stemming rules are applied, resulting to the TF-IDF vector for all the texts and their terms.

3.2 Categorization Procedure

The categorization subsystem is based on the cosine similarity measure, dot products and term weighing calculations. The system is initialized with a training set of 1500 pre-categorized articles, belonging to 7 different categories. The categorization

module receives as input the extract of the pre-processing mechanism, which is: a) stemmed keywords, b) noun-related information, c) absolute and relative frequency of the keywords appearance in the article and d) the article's title and body. After the initialization of the training set, the categorization module creates lists of keywords-nouns that are representative of a unique category, consisting of nouns with high frequency at a specific category, and small or zero frequency for the others.

The categorization attempt of a recently fetched article resembles the LLSF method and proceeds as follows; the labeling of the articles is done by using the list of the representative (stemmed) keywords of the text together with the frequencies evaluated by the pre-processing mechanism (Table 1). We then produce identical lists for all the categories that we own that consist of the same keywords followed by their frequency into the category (Table 2). In order to determine the text's category, we examine the cosine similarity of the text and the categories based on the aforementioned lists.

Table 1. Article's categorization vector

Stemmed k/w	Frequency
sharia	4
minist	7

Table 2. Politics category vector

Stemmed k/w	Frequency
sharia	0
minist	90

An article is most of the times related with a similarity measure to more than one category. However, for a categorization result to be accepted we define certain thresholds: (a) the cosine similarity between the text and the category should be over T_{hr1}, and additionally (b) the difference of the cosine similarity between the highest ranked category and the rest should exceed T_{hr2}. Experimentation, gave us the best suited thresholds for T_{hr1} and T_{hr2} as 0.50 (50% similarity), and 11% respectively. If T_{hr1} or T_{hr2} is not met, the article is forwarded to the summarization module and the resulting generic summary is used as input to a second categorization attempt for the article. Should the above thresholds be met, the labeling of the summary is kept, while at a different case, the initial labeling of the article is kept.

3.3 Summarization Procedure

During the summarization procedure, we utilize three factors: (a) the existence of a keyword in the title (b) the frequency of a keyword and (c) the noun tagging information of a keyword. We call these factors k1, k2 and N respectively. A keyword with very high frequency in the text is considered to be representative of it and thus, any sentence that includes it can be considered as text-representative. Additionally, any keyword of the text that also exists in the title is marked as an important one, so the sentences that include it are more representative. Moreover, when a keyword is tagged as a noun, we consider it significant thus boosting it with some extra weight. Parameters k1 and k2 are thoroughly explained in [1]. N derives from the following equation:

$$N = L * z \tag{1}$$

where z=0 if the keyword is not a noun and z=1 if it is. L conveys the desired extra weight that a noun existing in a sentence should have. Experimentation with various L values revealed that L should be no more than 1.5 or else sentences with few

keywords-nouns receive low scores, compared to sentences with many nouns, and are substantially excluded from the summary. Typical values for L range from 0 to 1 with the former depicting that the summarization algorithm is not taking into consideration the noun relevant information.

Based on these heuristics, we create a summary which consists of the most representative sentences of the text. In order to determine these, we deploy a score for each sentence according to the factors k1, k2 and N. Assuming that the text T has s sentences where i = [1..s] and f keywords where k = [1..f], each sentence is assigned a score according to the following equation:

$$W_i = \sum (1 + rel(fr(kw_{k,i})))(k_1 + k_2 + N) \tag{2}$$

where rel(fr(kwk,i)) is the relative frequency of the keyword k in sentence i.

After creating a generic summary, we retry to achieve a categorization, as the summarized text is more refined and consists only of important sentences rather than the whole text, which may include sentences with keywords that are distracting the categorization procedure.

The procedure that is followed in order to summarize a text after a successful categorization, differs from the aforementioned steps due to the fact that another factor is included in the scoring. This factor, namely k3 in [1], is the keyword's ability to represent the category to which the document belongs. As long as the text is categorized, we can utilize this factor in order to create a more efficient summary. With the use of k3, the overall weighting equation is depicted below.

$$W_i = \sum (1 + rel(fr(kw_{k,i})))(k_1 + k_2 + N)k_3 \tag{3}$$

4 Experimental Procedure and Results

In order to evaluate the summarization performance of PeRSSonal, with the appliance of noun retrieval techniques, we conducted two sets of experiments. Firstly, we tried to determine the best possible value for the L parameter of equation (1). Furthermore, we tried to evaluate the effect of the appliance of the noun retrieval algorithm explained earlier, to the overall system performance using classic IR measures. For conducting the experiments we utilized a corpus of 3000 news articles from various sources. The articles belonged with high relevance to one of the seven major categories of the system, and this information was used as explained at the previous section (precategorized articles) in order for the summarization procedure to produce the best possible summary.

As reported earlier, the parameter L is deployed for controlling the effect that noun retrieval has on the summarization procedure. We conducted experiments tuning L in order to decide on its best value as far as news articles, which is the case of PeRSSonal, are concerned. The various results are presented in Fig. 2.

At the previous graph it is clearly depicted that a value between 0.5 and 0.6 for L is best fitted. Values for L over 1 seem to attenuate both the precision and the recall of the summarization procedure compared to the L=0 case, i.e. when noun retrieval information is not used. This intuitively means that, when sentences that contain mostly

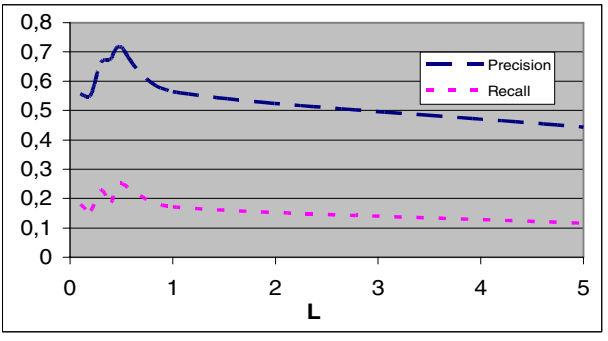

Fig. 2. Precision / recall results for summarization of news articles tuning the L value

nouns are kept at the summarization procedure, excluding the rest of the sentences, the effectiveness of the procedure slightly deteriorates. However, finding a golden section for the L parameter, which is dependable on the target texts, can enhance the summarization efficiency significantly. This is also obvious at the following graph where precision and recall results are depicted (using an L value of 0.6) when summarization proceeds with and without the noun-retrieval information.

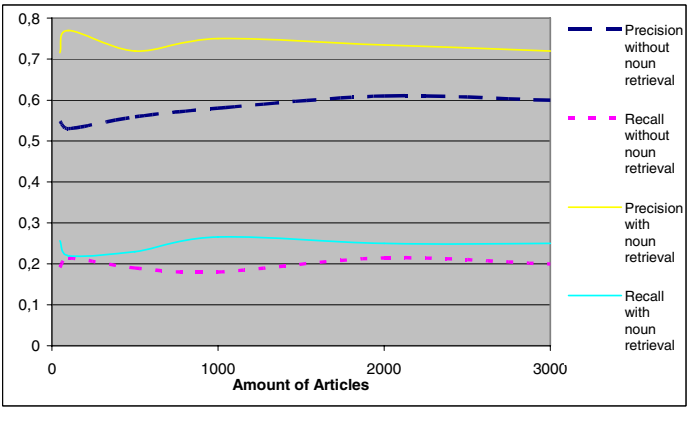

Fig. 3. Precision and Recall results for the PeRSSonal's summarization procedure when noun retrieval information is utilized and not

From Fig. 3 it is concluded that the noun retrieval information can give a notable precision boost to the resulting summaries compared to the case where noun retrieval information is not utilized; in other words, the resulting summaries are more precise. As far as recall is concerned, the improvement is small, yet significant, taking into account the fact that a text's summary represents a layer of abstraction, notably a low recall representation of the original text's information.

5 Conclusions and Future Work

In this paper we explored the effects that noun retrieval techniques, based on POS tagging, can have on information retrieval mechanisms and summarization in specific. Through the proposed framework that is utilized in an existing system, PeRSSonal, we are able to improve the summarization procedure by simple modifications to our keyword extraction algorithm. The efficiency improvements are small yet significant considering the fact that summarization is a difficult, mostly subjective procedure and that objective criteria of efficiency are difficult to appoint. Having incorporated noun retrieval techniques we are focusing on multilingual and multimedia support for PeRSSonal, the addition of which should require a throughout redesign of the main parts that consist the system. Also, we are considering a wider evaluation of the improvements that the applied noun-retrieval technique has on both the summarization and the categorization procedure.

References

[1] Bouras, C., Poulopoulos, V., Tsogkas, V.: PeRSSonal's core functionality evaluation: Enhancing text labeling through personalized summaries. Elsevier Science Publishers B. V, vol. 64, pp. 330–345. Elsevier, Amsterdam (2008)
[2] Wasson, M.: Using Leading Text for News Summaries: Evaluation Results and Implications for Commercial Summarization Applications. In: Proceedings of ICCL (1998)
[3] Goldstein, J., et al.: Summarizing Text Documents: Sentence Selection and Evaluation Metrics. In: Proceedings of ACM SIGIR Conference (1999)
[4] Ferragina, P., Gulli, A.: A Personalized Search Engine Based on Web-Snippet Hierarchical Clustering. In: Proceedings of WWW Conference (2005)
[5] Hayes, P.J., et al.: A News Story Categorization System. In: Proceedings of the second Conference on Applied Natural Language Processing (1988)
[6] Yang, Y., Chute, C.G.: An example-based mapping method for text categorization and retrieval. ACM Transaction on Information Systems (TOIS) 12(3), 252–277 (1994)
[7] Karlson, F., Voutilainen, A., Heikkila, J., Anttila, A.: Constraint Grammar. A Language-Independent System for Parsing Unrestricted Text. Mouton de Gruyter, Berlin (1995)
[8] Cutting, D., Kupiec, J., Pedersen, J., Sibun, P.: A practical part-of speech tagger. In: Proceedings of the 3rd Conference on Applied Natural Processing (1992)
[9] Roth, D., Zelenko, D.: Part-of-Speech Tagging Using a Network of Linear Seperators. In: Proceedings of the 36th Annual Meeting of the ACL – Coling, Mondreal, Canada (1998)
[10] Gimenez, J., Marquez, L.: SVMTool: A general POS tagger generator based on Support Vector Machines. In: Proceedings of the 4th International Conference on Language Resources and Evaluation, pp. 43–46 (2004)
[11] Bouras, C., Poulopoulos, V., Tsogkas, V.: The importance of the difference in text types to keyword extraction: Evaluating a mechanism. In: 7th International Conference on Internet Computing (ICOMP 2006), Las Vegas, Nevada, pp. 43–49 (2006)