Applying Graph-based Keyword Extraction to Document Retrieval

Youngsam Kim[1] **Munhyong Kim**[1] **Andrew Cattle**[1] **Julia Otmakhova**[1]
Suzi Park[1] **Hyopil Shin**[1]

[1]Seoul National University/ Gwanak-1, Gwanak-ro, Gwanak-gu, Seoul, South Korea
youngsamy@gmail.com, likerainsun@gmail.com, acattle@gmail.com,
julia.nixie@gmail.com, mam3b@snu.ac.kr, hpshin@snu.ac.kr

Abstract

This paper proposes a keyword extraction process, based on the PageRank algorithm, to reduce noise of input data for measuring semantic similarity. This paper will introduce several features related to implementation and discuss their effects. It will also discuss experimental results which showed significantly improved document retrieval performance with this extraction process in place.

1 Introduction

To date, the most popular and well known approach to calculating semantic similarity of text documents has been utilizing Vector Space Models (VSM). The key idea of VSM is to map each document in a corpus or collection into a vector of a vector space, and interpret the distance between the query document's vector and the other texts' vectors as their degree of semantic relatedness (Salton, 1971; Salton et al., 1975; Turney and Pantel, 2010).

The VSM evolved from the SMART information retrieval system (Salton, 1971) and SMART pioneered many important terms and concepts that were adopted by modern search engines (Turney and Pantel, 2010). Many search engines are reported to use VSM to calculate the similarity between a query and a document (Manning et al., 2008).

A common problem of VSM is that documents often contain words with high frequency but little semantic significance. VSM usually deal with this obstacle using tf-idf weighting and singular value decomposition techniques (Turney and Pantel, 2010).

In order to improve retrieval systems that use the models, we suggest that employing keyword extraction on a per document basis to help reduce the noise inherent in large texts. For this purpose, PageRank, a graph-based ranking algorithm, was used in this study. Graph-based analysis techniques represent a document or text as a graph consisting of nodes (terms or phrases) and edges (pre-defined relations). Along with Brin and Page (1998)'s PageRank, various modifications and other graph-based algorithms have been introduced and proved their usefulness in various natural language processing tasks (Erkan and Radev, 2004; Kurland and Lee, 2006; Mihalcea et al., 2004; Wang et al., 2007; Widdows and Dorow, 2002).

Graph-based extraction systems showed better performance over frequency-based systems on multiple-theme documents (Grineva et al., 2009). In this study, it was assumed that applications would benefit from being able to select important words from documents using the extraction system; not just for keyword extraction tasks, but also for any complex system that needs its input-data to be noise-reduced for future processes.

2 Theoretical Background

In this paper, typical core parts of VSM are applied to measure semantic similarity over documents. Therefore, instead of using raw frequencies tf-idf weighting is adopted and length normalization is performed on both queries and target documents (Salton and Buckley, 1988; Buckley, 2005). In addition, the traditional cosine similarity is used to calculate closeness scores between pseudo documents (queries) and documents (following Buckley, 2005). However, the vector space dimensionality reduction phase has been omitted to simplify the experiment process.

2.1 Representing Text as Graphs

Before applying the Graph-based approach, several preprocessing stages had to be implemented.

International Joint Conference on Natural Language Processing, pages 864–868,
Nagoya, Japan, 14-18 October 2013.

First of all, the words in the text were identified as vertices of the graph. In this study, only unigrams were considered as node candidates. These unigrams were then POS tagged and passed through a syntactic filter, which only allowed a particular subset of POS tags. Various syntactic filters were experimented with including nouns, verbs, and adjectives but the best results were obtained when only nouns were used.

2.2 Defining Relationship Between Vertices

The relationship between vertices was chosen to be a co-occurrence relationship. Two nouns of the text would be connected if they both occurred within a window of N pre-fixed words. N can be any integer from 2 to 10, but the number of vertices, V is always equal to or less than N because the words in the window must bear a relevant POS tag from a predefined set.

In English it is easy to determine which words are within a specific window since each word split by spaces usually corresponds to one POS tag. However, unlike English, Korean is an agglutinative language and most words consist of more than one morpheme, each with their own part of speech. The example below, (1) demonstrates this fact.

(1) 그 여자가 학교에 갔다.
Ku nyeca-ka hakkyo-ey ka-ss-ta
The(ku) woman(nyeca)-Normative(ka) school(hakyo)-Locative(ey) go(ka)-PST(ss)-FinalSuffix(ta)
'The woman went to school'

If the sentence above is POS tagged, the number of tagged members would be eight, three tags more than the English equivalent. What this means is that the average distance between particular POS tagged items in Korean sentences is longer than in English sentences. Presumably, this would lengthen optimal window size in Korean when compared to English. According to the related result of Mihalcea et al. (2004), the best performance was achieved with the window of 2. In our case, it is natural to assume the span of the window would be wider.

However, it is also possible to consider the segments divided by whitespaces as the candidate nodes for the graph, and perform POS tagging after the separation. In this way, we can disregard the distance between any lexeme and functional prefix/suffix attached to it. Under this scheme the normative case 'ka' in the middle is ignored and thus the words 'woman' and the 'school' in (1) have no distance between them.

This second approach is very similar to the way in which English text is translated as a graph, but it disregards information gained from grammatical relations between nouns and functional prefixes/suffixes glued to them. These two approaches were both experimented with and their results are compared in Section 4.1.

3 Experiment

3.1 System Framework

The components explained above were implemented in an integrated system, including text pre-processing, POS tagging, keyword extraction, term weighting, and finally calculating semantic similarity. The goal of this system was to retrieve semantically related documents using query documents from the collected corpus.

The general workflow of the system is presented in Fig. 1. The system is designed for easy addition or removal of any of the intermediate stages or processes for experimental purposes. Such configuration changes constituted the different experiment conditions of this research.

Text pre-processing indicates deletion of any special characters, emoticons, and foreign words to allow the sanitized text to be parsed safely during part of speech analysis. And the POS tagger assigns each morpheme one of 22 tag sets and the words given the tag of 'noun', excluding pronouns, are passed for later processing.

To establish the stop-word list, three reviewers examined all the nouns extracted from the 800 documents that were collected for this study and selected 192 lexical items manually only if at least two of the reviewers voted for the same word to be on the list.

The Graph-based Keyword Extraction, Frequency Counting, and tf-idf Weighting modules may vary or be absent across various experiment conditions (This is denoted by the dotted boxes in Fig. 1). Keyword extraction stage always follows after Graph analysis because it is the process for sorting and choosing the adequate number of keywords. If the graph analysis is not performed, there can be no keyword extraction.
The Graph-based Keywords Extraction and Frequency Counting are mutually exclusive and only one method is chosen for each experiment condition.

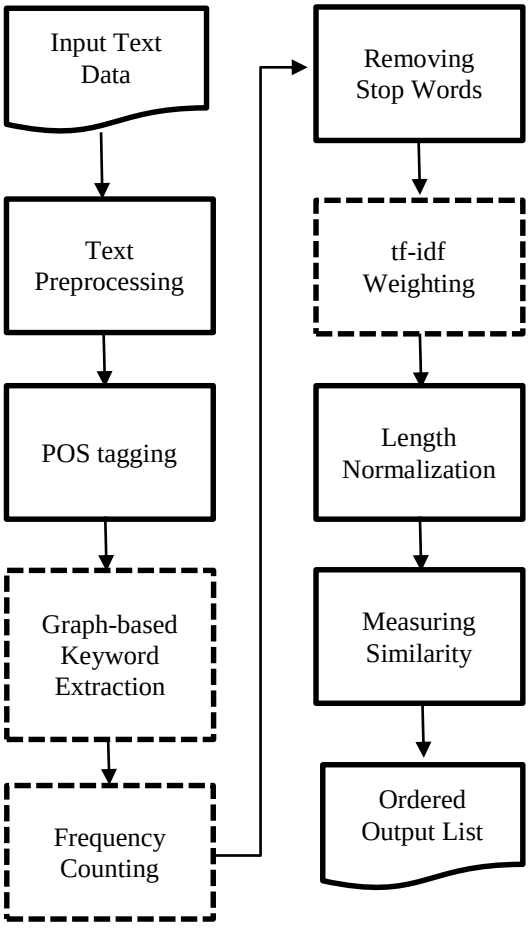

Figure 1. System workflow

3.2 Experiment Conditions

Three conditions in total were examined using the system. The first model, using graph-based ranking algorithm, adopted the tf-idf weighting method but omits the Frequency Counting stage. As such, the tf-idf module was supplied with the weighted values given by the Graph-based Keyword Extraction module as instead of the results of the Frequency Counting module. In this circumstance, tf and idf would not stand for term-frequency and inverse-document-frequency, but instead for term or inverse-document weighted scores. This condition can be regarded as a kind of integrated model of graph analysis and VSM.

The second model differed from the first in that the tf-idf weighting module received the frequencies for the keywords from the Frequency Counting module. This type of integration allowed the term-document matrix to be constructed in the way typical of tf-idf systems but the number of the rows was reduced because, same as in the first model, only a subset of the terms were selected during the Keyword Extraction process.

The final experiment condition was a typical form of VSM and provided a control measure of semantic similarity by using the traditional method, as described in Section 2. This model did not implement the graph-ranking algorithm and skipped the Keyword Extraction stage.

3.3 Data

To collect the test data set, 30 famous objects on a list of Seoul cultural assets, each including some descriptive text, were selected for use as queries. For each query, 20 related documents were manually collected including Wikipedia documents, blog posts, and news articles written on the object. Two hundred texts unrelated to any of the queries were searched and stored. These texts consisted of articles, blogs, and web page texts on various topics but limited to social, economic or cultural contents. Hence, each of the cultural objects there would be 20 semantically related documents against 780 unrelated texts.

3.4 Evaluation Scheme

To estimate the performance of the models in this system, the well-known measure Mean Average Precision was used (Voorhees and Harman, 2005). This measure ranges from 0 to 1 where the maximum value 1 means that all target documents are placed higher than the non-related texts in the ordered output list.

4 Results

4.1 Morpheme vs. Word

For the 30 queries, given a word-window of 4 and a Proportion of Keywords of 0.4, the mean average of the precision (MAP score) for the morpheme-based criteria was 0.83 while the word-based criteria was 0.74.

Only four of the 30 queries showed higher MAP scores for the word-based separation method. Thus, in this study, the morpheme-based approach significantly outperforms the word-based approach.

4.2 Window Size

A Determining the length of word-window is related to the problem of morpheme/word based separation. In some languages, especially agglutinative languages like Korean, one word might be composed of more than three morphemes. Practically, this means that if there are two words both containing a noun and split by whitespace,

the probability of finding a morpheme between of them will be higher than in non-agglutinative languages such as English, widening the mean distance between any two nouns.

To confirm this prediction, an experiment manipulating the size of the window was conducted and the result is presented in Fig. 2. As one can see, the highest the MAP value (0.83) for the morpheme-based split method is obtained with a window length of 4. This pattern is what was expected given the discussion in Section 2.2. In contrast, using the word-based separation method, there does not seem to be any significant relationship between window size N and MAP score.

One easy interpretation of this result is that the word-based solution is not effective enough to capture the connective pattern of the terms in the network since it is missing the syntactic cues associated with words' stems.

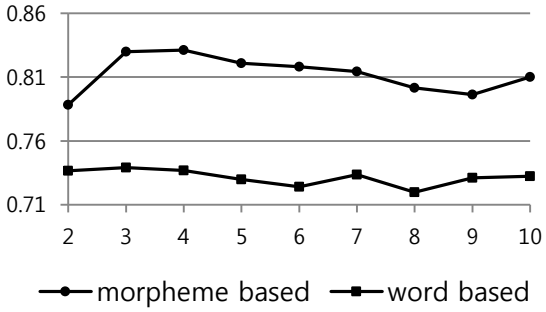

Figure 2. Mean average precision score in terms of window of word N.

4.3 Proportion of Keywords

In Fig. 3, the window size was set to 4 and the highest performing experiment model was used (the comparison result for the different models will be provided in next section).

As one can see in the graph, a proportion of 4/10 recorded the highest score. However, after this point the MAP score rapidly dropped; in contrast with the period of gradual increase observed up to that point.

To understand this result, it is important to recall that the tf-idf weighting mechanism uses inverse document frequency to give weights to the terms that are found in only a small number of documents but are frequent within a particular document. Hence, removing too many terms from the text would artificially increase the value of the idf component of tf-idf as a word may be rarely selected as a keyword despite occurring in a large number of documents.

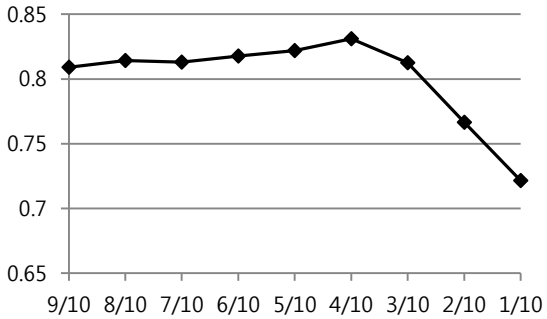

Figure 3. Mean average precision scores for the change of the keyword proportion (e.g., 9/10 means that 90 percent of the candidate nouns were accepted).

4.4 Comparison of Different Models

In Table 1, the notation [+/−] indicates whether the function was employed in the workflow of the experiment. Thus, Keyword extraction with plus sign means that the PageRank-based ranking algorithm was used for the keyword extraction process. Similarly, if the tf-idf module is displayed with minus sign then it means that frequencies per each term in the documents were replaced with the values from the graph-based analysis module.

Model	MAP score
Keyword extraction +, tf-idf −	0.81
Keyword extraction −, tf-idf +	0.80
Keyword extraction +, tf-idf +	**0.83**

Table 1. The MAP scores for different experiment conditions.

The results in Table 1 reveal that the full model (including all the sub-modules) outperforms the other two models, proving the research assumption that pre-filtering input texts would improve the quality of semantic similarity measurements based on VSM. The other modified condition employing the graph-analysis recorded the second highest, but the difference to the control condition was very small. These results were obtained using a window size of 4 and a keyword proportion of 4 out of 10; these values yielded the best outcome from the experiments.

In short, the comparative result of the experimental conditions suggest that drawing a bag of filtered words per document before tf-idf weighting could improve the process of computing semantic relatedness.

Acknowledgments
The authors would like to thank the three anonymous reviewers for their helpful and valuable comments.

References

Brin, Sergey, & Page, Lawrence. (1998). The anatomy of a large-scale hypertextual Web search engine. Paper presented at the Proceedings of the seventh international conference on World Wide Web 7, Brisbane, Australia.

Buckley, Chris. (2005). Project at TREC. In Ellen M. Voorhees & Donna K. Harman (Eds.), TREC : experiment and evaluation in information retrieval (pp. 301-320). Cambridge, Mass.: MIT Press.

Erkan, Gunes, & Radev, Dragomir R. (2004). LexPageRank: Prestige In Multi-Document Text Summarization. Paper presented at the Conference on Empirical Methods in Natural Language Processing, Barcelona, Spain.

Grineva, Maria, Grinev, Maxim, & Lizorkin, Dmitry. (2009). Extracting key terms from noisy and multitheme documents. Paper presented at the Proceedings of the 18th international conference on World wide web, Madrid, Spain.

Kurland, Oren, & Lee, Lillian. (2006). Respect my authority!: HITS without hyperlinks, utilizing cluster-based language models. Paper presented at the Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, Seattle, Washington, USA.

Manning, Christopher D., Raghavan, Prabhakar, & Schtze, Hinrich. (2008). Introduction to Information Retrieval: Cambridge University Press.

Mihalcea, Rada, & Tarau, Paul. (2004). TextRank: Bringing Order into Texts. Paper presented at the Conference on Empirical Methods in Natural Language Processing.

Salton, Gerard M. (1971). The SMART Retrieval System: Experiments in Automatic Document Processing: Prentice-Hall, Inc.

Salton, Gerard M., & Buckley, Christopher. (1988). Term-weighting approaches in automatic text retrieval. Inf. Process. Manage., 24(5), 513-523. doi: 10.1016/0306-4573(88)90021-0

Salton, Gerard M., Wong, Andrew, & Yang, ChungShu. (1975). A vector space model for automatic indexing. Commun. ACM, 18(11), 613-620. doi: 10.1145/361219.361220

Turney, Peter D. (2006). Similarity of Semantic Relations. Comput. Linguist., 32(3), 379-416. doi: 10.1162/coli.2006.32.3.379

Turney, Peter D., & Pantel, Patrick. (2010). From frequency to meaning: vector space models of semantics. J. Artif. Int. Res., 37(1), 141-188.

Voorhees, Ellen M., & Harman, Donna K. (2005). Project at TREC. Cambridge, Mass.: MIT Press.

Wang, Jinghua, Liu, Jianyi, & Wang, Cong. (2007). Keyword extraction based on pagerank. Advances in Knowledge Discovery and Data Mining, 857-864.

Widdows, Dominic, & Dorow, Beate. (2002). A graph model for unsupervised lexical acquisition. Paper presented at the Proceedings of the 19th international conference on Computational linguistics - Volume 1, Taipei, Taiwan.