2005－NL－170（20）
2005／11／22

情報検索のために単一ドキュメントからキーワード抽出

デービット・ブレスウェル　任　福継　黒岩　眞吾
知能情報工学科　徳島大学
〒770-8506 徳島市南常三島 2-1

情報検索の基本要素となるキーワードは、ドキュメントの探索から記述にわたってあらゆることに使われている。典型的に、キーワード抽出のアルゴリズムでは、キーワード抽出するため、ドキュメントの収集が必要とされる。ドキュメント収集なしのキーワード抽出は重要性を獲得することである。この問題に関しては既に研究されている。しかし、二つの難題が残されている。一つは、キーワードの質は情報検索作業でどれほど機能するかという点に基づいていないのである。もう一つは、キーワードは一つの言語に特定されているのである。本稿では、多言語に適用でき、しかも、有効的にキーワードを抽出できる新しいアルゴリズムを提案した。

Keyword Extraction from a Single Document for Information Retrieval

David B. Bracewell, Fuji REN, and Shingo KUROIWA
Department of Information Science and Intelligent Systems,
Faculty of Engineering, The University of Tokushima
Tokushima 770-8056

Keywords are a fundamental part of information retrieval. Keywords are used for everything from searching to describing a document. Typically, algorithms for keyword extraction require a document collection in order to extract keywords. Extracting keywords without a document collection is gaining importance. Research has been done to deal with the problem. However, there are two problems 1) the quality of the keywords was not based on how well they perform in IR tasks and 2) they were designed for only one language. This paper proposes a new algorithm that is applicable to multiple languages and extracts effective keywords.

1. INTRODUCTION

Keywords or index terms are a fundamental part of information retrieval. They describe an entire document in a few words and have uses in document retrieval, web searches, text categorization, text summarization, etc. As such, effective keywords are a necessity. Typically, when keywords are generated they are done so having the entire document collection at hand. This is to ensure that the keywords chosen are, for the most part, independent or in other words that the chosen keywords appear frequently in one document but not in the rest of the documents. Such techniques tend to work well to identify the documents, but require the entire collection to be determined and collected beforehand.

When it comes to extracting keywords from a single document only, the task becomes harder. The reason is that keywords between documents in a collection are expected to be independent when it comes to information retrieval tasks. When keywords are generated from just a single document there is no guarantee that they will be independent of those from other documents. Methods such as co-occurrence [2], [3] and machine learning [4], [5] have been used for extracting keywords from single document.

In the previous work the authors evaluated their algorithms using human judges or examine the agreement, most often measured in precision and recall, between the algorithm's keywords and those created by humans. This may well be a proper evaluation if the goal is to try to mimic humans or make keywords that humans will use. However, if this is not the case and the keywords will be used by other information retrieval algorithms then human judged or agreement with human keywords is not a useful form of evaluation. In addition, these algorithms were designed for and tested on only one language. The applicability to other languages is unknown. In contrast, the evaluation method used in this paper uses an information retrieval task, keyword search, to determine the effectiveness of the keywords at uniquely describing the document. Also, testing is done on English, Japanese and Chinese documents so that the applicability to other languages can be seen. In a later section of this paper the related work will be examined in more detail.

We are in the early stages of building a cross-lingual information extraction, management and presentation system. As such, a keyword extraction algorithm that is able to handle many languages is needed. Using linguistic information is helpful in many IR tasks. Because of this, the algorithm presented in this paper needs certain language dependent components, a morphological analyzer and simple noun phrase (NP) grammar, in order to determine the keywords. However, these components are readably available for most languages. Moreover, there are many algorithms that are able to be used with languages that they were not designed for, for example the TnT Tagger [1] is capable of dealing with any language that segments words with whitespace. The algorithm in this paper will be used in various ways in the system, spanning from topic analysis and summarization to finding relevant or similar documents. As such, how humans perceive the keywords or how the keywords agree with human created ones is less of a concern than how well the keywords uniquely and accurately describe the document they were extracted from.

The paper will proceed as follows, in section 2; background information about the system that is being developed is given. Next, in section 3 the corpus will be briefly discussed. In section 4 the algorithm will be explained. Experimental results will be shown in section 5. In section 6 related work will be discussed. Finally, in section 7 concluding remarks and future works will be discussed.

2. BACKGROUND

At present we are stating to build a system entitled KANT (Knowledge Acquisition, iNterpretation, and Translation.) The focus of the KANT system is information extraction, management, and presentation in a multilingual environment. The goal is to allow a user to search for information in their native language and retrieve answers from texts in any language. To the user the process is transparent and they would not know what the original language was for the answer/information they obtained.

The system will be designed to be used in dealing with news and educational topics (history, literature, arts, music, etc.) For the time being Japanese, English, and Chinese are the targeted languages. The 4 major components of KANT are:

1. Knowledge Acquisition
2. Knowledge Interpretation
3. Knowledge Translation
4. Knowledge Presentation

The Knowledge Acquisition module is made up of keyword extraction and restricted domain creation. The Knowledge Interpretation module is made up of named entity recognition and topic analysis. The Knowledge Translation module is what allows for knowledge in one language to be acquired, interpreted, or presented in another. Finally, the Knowledge Presentation module is made up of question & answering, report generation, and text summarization.

The first step in this system is to build an effective keyword extraction algorithm. Since the underlying knowledge base is not able to be collected beforehand typical keyword extraction algorithms may not be suitable. Moreover, the previous methods for keyword extraction from single documents were more focused on extracting keywords that humans would agree with than they were with extracting keywords that are useful in IR tasks. Because of these reasons and the fact that the extracted keywords will be play a pivotal role in the system as they will be used by all the other modules, a new algorithm was needed.

3. CORPUS INFORMATION

The Japanese-English bilingual corpus that was used [6] was designed based on the idea proposed by Resnik et al [7] of mining the web for bilingual databases. The corpus is regularly updated and currently has over 17,000 document pairs. The documents come largely from Wired (http://www.wired.com) whose articles are translated daily into Japanese. While the main topic is technology news there are many subtopics. Henceforth, this corpus will be referred to as the Wired corpus.

For Chinese, news articles were mined from Yahoo! China News (http://cn.news.yahoo.com/) over a week long period. Articles were taken from every category and should provide a good base set for evaluation. Henceforth, this corpus will be referred to as the Yahoo corpus.

4. KEYWORD EXTRACTION ALGORITHM

The keyword extraction algorithm was designed to be as language independent as possible. As such, it was broken up into 3 major modules and each either uses no language-dependent components at all or components that are commonly available for most languages. The 3 modules of the algorithm are given below.

1. Morphological Analysis
2. Noun Phrase (NP) Extraction and Scoring
3. Noun Phrase (NP) Clustering and Scoring

Each of these steps will described in detail in the following subsections. Figure 1 shows a pictorial representation of the algorithm. While the goal of the algorithm is to create a good set of keywords, we also want to make sure the algorithm's performance, in terms of speed, is high.

4.1 Morphological analysis

Morphological analysis is the identification of word stems and, optionally, syntactic categories (Parts-of-Speech). It is a fundamental part of Natural Language Processing (NLP). As such, it easy to find morphological analyzers for most languages. During the morphological analysis portion of the algorithm stemming, part-of-speech tagging, and word segmentation is performed. After morphological analysis unigram word frequencies are collected from the document.

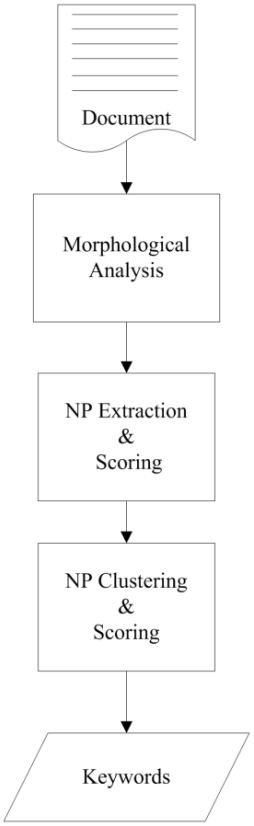

Fig. 1. Algorithm Overview

These frequencies will later be used to help score the noun phrases. They are collected at this part of the algorithm due to the fact that certain languages, such as Japanese and Chinese, need to be segmented first. This module's process is shown in the order of execution below.

1. Word Segmentation
2. Part-of-Speech Tagging
3. Stemming
4. Unigram Frequency Calculation

For English, Porter's stemmer [9] and Brill's tagger [10] were used. These two algorithms are standards in the NLP community. Brill's tagger was chosen, because it gives relatively good results and its performance, in terms of speed, is acceptable. For Japanese, meCab was used [11]. meCab was chosen due to its speed improvements over other Japanese morphological analyzers. For Chinese, the Stanford Parser [12], which can also be used as part-of-speech tagger, was used as it was the easiest to use tool that we could find that was capable of dealing with Chinese.

4.2 NP extraction and scoring

The keywords are restricted to only being noun phrases as noun phrases often contain the most important information. For both Japanese and English a simple NP chunking algorithm based on a Context Free Grammar (CFG) was used. The English CFG was based on the simple NP Augmented Transition Network (ATN) in [13]. The Japanese CFG was handmade and provides chunking of simple noun phrases. Such simple CFGs should be able to be easily made for every language. In fact for the purposes of keyword extraction the noun phrases can be very simple and it may be possible to create a template CFG for noun phrases. For each new language the only thing that would need to be changed from the template would perhaps be the word order. In this paper we used a simple CFG to chunk noun phrases, however, a parser could have been used instead. The reason it was not used here is to keep the execution time of the algorithm low.

After the noun phrases are marked and extracted from the article, stopwords are removed. Next, the frequency of each noun phrase is calculated and used in the scoring method. The scoring method takes into account the unigram frequency of the individual words in the noun phrase and the frequency of the noun phrase. Equation 1 shows the calculation of Unigram Frequency for an NP, which is simply the summation of unigram frequencies of the individual words. Equation 2 shows how to calculate the score for an NP. In this equation NPF(NP) refers to the frequency that the noun phrase occurred in the article and |NP| means the number of words in the NP.

$$UF(NP) = \sum_{i=0}^{|NP|} UnigramFrequency(w_i) \quad \textbf{(1)}$$

$$Score(NP) = \frac{UF(NP) * NPF(NP)}{|NP|} \quad \textbf{(2)}$$

4.3 NP clustering and scoring

After the noun phrases are scored they are clustered. The clustering is an attempt to prevent redundancy in

the extracted keywords. For example, in a document there may exist noun phrases such as ”stem cell” and ”stem cell research” that have large NP scores. Since they deal with the same general topic and choosing both of them as keywords may cause a lesser scored, but equally important, noun phrase from being included, clustering is done.

In order to be language independent, the clustering process is very simple. Two noun phrases are said to be in the same cluster if they have a word in common. The clustering algorithm starts by first by assigning all one word NPs to their own cluster. Then, multi-word NPs are assigned to every cluster that they share a word in common with. Finally, any non-assigned multi-word NPs are assigned to their own cluster. An example of what the resulting clusters make look like can be seen in figure 2. While semantic similarity could be used, such as it was in [14], not every language may have the tools necessary to use such an approach. For this reason and for speed considerations this simple method is employed.

After the noun phrases are clustered they are scored. A cluster’s score is the average NP score of the noun phrases in the cluster, as seen in equation 3.

$$Score(Cluster) = \frac{\sum_{i=0}^{|Cluster|} Score(NP_i)}{|\,Cluster\,|} \quad (3)$$

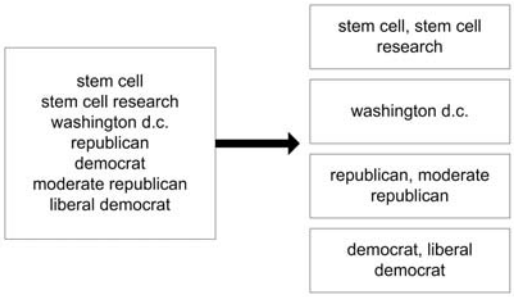

Fig. 2. Clustering Example

4.4 Choosing keywords

In choosing the keywords the clusters are sorted by score. Centroids from the top N scoring clusters are chosen to be the keywords for the article. The centroid of a cluster is the shortest word in the cluster. In future, we will look at using the highest scored NP out of the cluster.

5. EXPERIMENTAL RESULTS

For comparison purposes a baseline algorithm was created. This algorithm simply removes stopwords, does stemming, and returns the top N most frequent words. In testing the validity of keywords there are two methods:

1. Agreement with human generated keywords.
2. Results of using keywords in some Information Retrieval (IR) task.

The benefit of the first method is that if there is a high agreement then we know that the generated keywords are generally acceptable to humans as being an accurate set. The problem, however, is that human created keywords are subjective and two different people will generate two different sets of keywords. Another problem is that often humans create keywords that do not appear in the document. In this case it would be very hard, if not impossible, for an algorithm to agree with the human. The second method has the benefit that it is easier to evaluate. Also, since the keywords generated by this algorithm are intended to be used later on in IR tasks it makes more since to see how well they perform in one. The downside of this method is that since there is no checking with human generated keywords it can not be known for certain if the extracted keywords are acceptable to humans as an accurate set of keywords for the document. However, if the extracted keywords are not to be used by humans then this is not a problem.

In this paper the second method was used. The effectiveness of the keywords in searching for the document they were extracted from was evaluated. For searching, Rel [15] was used. Rel is a suite capable of doing information retrieval on full text using multiple keywords with boolean operators. It uses word incident rate to determine the relevance of a document to a set of keywords.

5.1 English Results

5,000 randomly selected articles from the Wired corpus were used. The algorithm was set to extract 10 keywords. Table 1 shows the search results. In the table, ”In top 10” means the document the keywords were generated from was in the top 10 most relevant documents, ”In Top 3” means it was in the top 3 most relevant documents, and ”#1 Result” means it was the most relevant document.

Algorithm	In Top 10	In Top 3	#1 Result
Baseline	30.23%	29.53%	23.63%
Proposed	96.82%	96.30%	93.68%

Table 1. English Results

The proposed algorithm is a vast improvement over

the baseline method. The correct document was the number on result in the search 94.68% of the time compared to just 26.63% of the time for the baseline algorithm. This shows that simply using unigram frequencies of all words is not acceptable and that noun phrases work well as keywords.

5.2 Japanese Results

As with the English evaluation, 5,000 randomly selected articles from the Wired corpus were used. The algorithm was set to extract 10 keywords from the document. Table 2 shows the results.

Algorithm	In Top 10	In Top 3	#1 Result
Baseline	96.70%	89.49%	77.02%
Proposed	99.90%	99.43%	96.62%

Table 2. Japanese Results

The results show that, like with English, the proposed algorithm outperforms the baseline algorithm. Both the baseline and proposed algorithm performed better for Japanese than English. This is can be explained by the fact that the Japanese use of Kanji (Chinese characters) helps, in a small part, to disambiguate the words possibly making the keywords more distinct.

5.3 Chinese Results

For Chinese, 2000 articles from the Yahoo corpus were randomly chosen for evaluation. The algorithm was set to extract 10 keywords from the document. Table 3 shows the results.

Algorithm	In Top 10	In Top 3	#1 Result
Baseline	94.30%	85.15%	75.25%
Proposed	99.10%	98.00%	95.95%

Table 3. Chinese Results

The results are similar to those of English and Japanese. The proposed algorithm outperforms the baseline algorithm in extracting keywords that uniquely identify a document. As with Japanese, the Chinese characters help in some part to disambiguate words causing the baseline and proposed algorithms to perform better than English.

6 RELATED WORK

Generally in information retrieval keywords are generated for documents from a predefined document set. The most notable method for doing this is to use TF-IDF (Term Frequency-Inverse Document Frequency) weighting[16]. This method extracts keywords that occur frequently in one document and infrequently in the rest of the document collection. While this method works well the document collection must be defined beforehand. Using this method to extract keywords from just a single document, when a document collection is not given, should yield results similar to the baseline algorithm presented earlier. The reason is that since there are no other documents to compare the keywords to the algorithm will simply choose keywords based on term frequency.

One common approach to the problem of extracting keywords from a single document is using machine learning. Examples of such are [5]. Hulth looked at using linguistic data with supervised machine learning algorithms [4]. She looked at using different classifiers and different features to train on. As with this paper, she found that NP Chunking was a good approach. Since this method is supervised learning it requires a manually annotated corpus. Each domain will require that the classifier be retrained on a new manually annotated corpus. This would prove cumbersome if dealing with an open domain or a set of multiple domains. Evaluation was done by measure the precision and recall with human generated keywords so the suitability of the keywords for IR tasks is unknown.

Matsuo and Ishizuka looked at using co-occurrence information for extracting keywords from a single document [2]. Like most previous research they used human judges for evaluation. They also had a rather small test set of papers from only 20 authors. Because of these reasons it is unknown if this method would be suitable for IR tasks.

The previous approaches for extracting keywords from a single document have some problems for the system that we are developing. One is that the effectiveness of the algorithm was left up to human judges. This is a problem as it is highly unlikely for any 2 persons to agree on a set of keywords. Therefore, the results that are given are subjective and may not reflect the true effectiveness of the keywords. Moreover, since the system will use the keywords and not humans then evaluations of the effectiveness of the keywords should be done using an IR task. Since this was not done the keywords extracted by such algorithms may not be effective in the system. The second problem is that the algorithms were only evaluated using one language. While they may well work with other languages it is not a certainty.

7 CONCLUSION AND FUTURE WORK

This paper presented a multi-language capable algorithm for keyword extraction from a single document for information retrieval. The only requirement the algorithm has for a language is that the language have a morphological analyzer and rules for finding simple noun phrases. Since nouns contain

the bulk of information, noun phrases are extracted and become candidate keywords. The noun phrases are scored and clustered and then the clusters are scored. The shortest noun phrases from the highest scoring clusters are then used as the keywords.

The algorithm was tested using an English and Japanese bilingual corpus and small Chinese news corpus. The results showed that the algorithm performed better than the baseline for all the languages. It was also shown that the algorithm extracts keywords that are effective at uniquely describing the document. We also found that the Japanese and Chinese keywords did slightly better than English ones.

In the future we will use the keywords in other IR tasks. They will be used to compute the similarity between documents and to automatically create restricted/special domains. They will also be useful in question & answering, text summarization, topic analysis, and report generation.

ACKNOWLEDGMENT

This research has been partially supported by the Ministry of Education, Culture, Sports, Science and Technology of Japan under Grant-in-Aid for Scientific Research (B), 14380166, 17300065, Exploratory Research, 17656128, 2005.

REFERENCES

[1] T. Brants, ”TnT – a statistical part-of-speech tagger,” In Proceedings of the 6th Applied NLP Conference (ANLP-2000), 2000.

[2] Y. Matsuo and M. Ishizuka, ”Keyword Extraction from a Single Document using Word Co-occurrence Statistical Information,” International Journal on Artificial Intelligence Tools, vol.13, no.1, 157-169, 2004.

[3] Y. Ohsawa, N. E. Benson, and M. Yachida, ”KeyGraph: Automatic indexing by cooccurrence graph based on building construction metaphor,” In Proceedings of the Advanced Digital Library Conference, 12, 1998.

[4] A. Hulth, ”Improved Automatic Keyword Extraction Given More Linguistic Knowledge,” In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP’03), 2003.

[5] P. Turney, ”Learning algorithms for keyphrase extraction,” Information Retrieval, vol. 2, no. 4, 303-336, 2000.

[6] J. Fry, ”Parallel Japanese-English corpus,” [Online], Available at http://johnfry.org/blog/.

[7] P. Resnik and N. A. Smith, ”The Web as a parallel corpus,” Computational Linguistics, vol. 29 , no. 3, 349 - 380, 2003.

[8] D. D. Lewis, Y. Yang, T. Rose, and F. Li, ”RCV1: A New Benchmark Collection for Text Categorization Research,” Journal of Machine Learning Research, vol. 5, 361-397, 2004.

[9] M.F. Porter, ”An algorithm for suffix stripping”, Program, 14(3), 130-137, 1980.

[10] E. Brill, ”A Simple Rule-Based Part-of-Speech Tagger,” in Proceedings of 3rd Applied Natural Language Processing, 152-155, 1992.

[11] T. Kudo, MeCab: Yet Another Part-of-Speech and Morphological Analyzer, [Online], Available at http://chasen.org/ taku/software/mecab/.

[12] Stanford NLP Group, The Stanford Parser, [Online], Available at http://nlp.stanford.edu/downloads/lex-parser.shtml.

[13] T.Winograd, Language as a Cognitive Process Volume 1: Syntax, Reading, Massachusetts: Addison-Wesley, 1983.

[14] D.B. Bracewell, S. Russell, A.S. Wu, ”Automatic Identification, Expansion, and Disambiguation of Acronyms in Biomedical Texts,” To Appear in Proceedings of the 3rd International Symposium on Parallel and Distributed Processing and Applications, 2005.

[15] NFormatiX, Software For Full Text Information Retrieval Over The Internet, [Online], Available at http://www.johncon.com/nformatix/.

[16] G. Salton and C. Buckley, ”Term Weighting Approaches in Automatic Text Retrieval,” Information Processing and Management, vol. 24, no. 5, 513 -523, 1988.