Improved Automatic Keyword Extraction Based on TextRank Using Domain Knowledge

Guangyi Li and Houfeng Wang

Key Laboratory of Computational Linguistics, Ministry of Education,
Institute of Computational Linguistics, School of Electronics Engineering and Computer Science, Peking University
{liguangyi,wanghf}@pku.edu.cn

Abstract. Keyword extraction of scientific articles is beneficial for retrieving scientific articles of a certain topic and grasping the trend of academic development. For the task of keyword extraction for Chinese scientific articles, we adopt the framework of selecting keyword candidates by Document Frequency Accessor Variety(DF-AV) and running TextRank algorithm on a phrase network. To improve domain adaption of keyword extraction, we introduce known keywords of a certain domain as domain knowledge into this framework. Experimental results show that domain knowledge can improve performance of keyword extraction generally.

Keywords: Keyword Extraction, TextRank, Domain Knowledge.

1 Introduction

Keywords, consisting of one single word or several words, summarize topics and ideas of an article. Keywords can benefit many NLP applications, such as text categorization, document clustering, search engine, etc. In an era when information on Internet grows explosively, it is intractable to scan every document thoroughly. Keywords enable us to find documents we need from the ocean of information.

In order to capture the topics of an article accurately and sufficiently, keywords usually need to be assigned by experts with adequate domain knowledge. However, with innumerous documents emerging everyday, it would be too costly to assign keywords to documents by human efforts. Therefore, automatic keyword extraction is drawing interests of many researchers and a number of techniques are applied to this task successfully.

The targets of keyword extraction include news articles, web pages, scientific articles, etc. Study of keyword extraction for scientific articles is getting more attention recently, since keywords are essential for retrieving scientific articles and grasping the trend of academic development. Though keywords are usually required for scientific articles and academic dissertations, many authors have trouble selecting proper keywords. And different authors will give keywords following different criteria An efficient keyword extraction system can aid authors

C. Zong et al. (Eds.): NLPCC 2014, CCIS 496, pp. 403–413, 2014.

in selecting proper keywords and help to correct inadequate keywords given by authors.

For keyword extraction for scientific articles, a shared task was held at the Workshop on Semantic Evaluation 2010 (SemEval-2010 Task 5) [1]. It was geared towards scientific articles in English. However, research on keyword extraction towards scientific articles in Chinese is relatively rare. We extract keywords from Chinese scientific articles adopting the framework of selecting keyword candidates by Document Frequency Accessor Variety(DF-AV) and running TextRank algorithm on a phrase network. We show how to improve the result of unsupervised keyword extraction for a certain domain with domain knowledge of known keywords.

The remainder of the paper is organized as follows. Section 2 reviews related work. Section 3 describes the general framework for keyword extraction. Section 4 shows why known keywords can be used as domain knowledge and how to improve the result of keyword extraction with domain knowledge. Section 5 represents our experiment results and Section 6 concludes our work.

2 Related Work

The task of keyword extraction is usually divided into two steps: candidate selection and keyword ranking. Most keywords are nouns or noun phrases. Therefore, for candidate selection, most work is based on n-gram [2–4] or Part-of-speech tags [5, 6] or both [7]. Especially, [8] compared n-gram, POS tags and NP-chunks and demonstrated that voting from the three methods performs best.

Choosing from keyword candidates is usually considered a ranking problem. Each candidate is assigned a score, and top-k ranked candidates are chosen as keywords. Statistics are commonly used feature for ranking, among which TF-IDF is the most popular feature [2, 6, 9, 10]. Word co-occurrence is another widely used feature [11–13]. Supervised learning methods are also adopted for keyword extraction, including maximum entropy [6], naive Bayes [2, 12],support vector machines [14], conditional random field [15], etc. [16] gives a summary of systems which participated in SemEval-2010 Task 5. The best performance is achieved by bagged decision tree [17].

TextRank [18] is a graph-based, unsupervised ranking algorithm. It performs well for keyword extraction and becomes popular recently. Related research includes [19], [20], etc.

3 General Framework

In this section, we describe the framework based on TextRank for keyword extraction for Chinese scientific articles. First, keyword candidates are selected from the document by Document Frequency Accessor Variety(DF-AV). Second, we build a phrase network using candidates and rank candidates with TextRank. Top-k ranked candidates are selected as extracted keywords.

3.1 Candidate Selection by DF-AV

Keywords of scientific articles are mostly noun phrases. As for English, defined POS sequences are used to select keyword candidates. However, for Chinese, this might not work well, as accuracy of POS tagging for Chinese scientific articles is not satisfactory. There are two main reasons. First, Chinese words have fewer morphological changes than English. For instance, verb "extract" and noun "extraction" will be translated to the same Chinese word. This brings difficulty to Chinese POS tagging. Second, most Chinese POS tagging systems are trained on news corpus, while many keywords of scientific articles rarely appear in news corpus, i.e., these words are Out-of-Vocabulary words. Therefore, POS tagging for Chinese words may contain many errors.

As a consequence, We use the statistical criterion instead of POS sequence to select keyword candidates. Accessor Variety(AV) [21] is a statistical criterion first used for new word extraction from Chinese text collections. The criterion is proposed from the viewpoint that a word is a distinguished linguistic entity which can be used in many environments. Therefore, the numbers of different characters appearing before and after a word is relatively high. Likewise, we can adopt Accessor Variety for keyword candidates selection.

For a phrase string phr, let S_L denote the set of words appearing before phr, S_R denote the set of words appearing after phr. Thus, left Accessor Variety of phr $AV_L = sizeof(S_L)$, and right Accessor Variety of phr $AV_R = sizeof(S_R)$. The larger Accessor Varieties are, the more likely phrase phr is a keyword candidate. We define score of phrase phr as $Score(phr) = Freq(phr) \times AV_L(phr) \times AV_R(phr)$. All phrases whose score is higher than a certain threshold are selected as keyword candidates.

However, criterion of Accessor Variety cannot deal with low-frequency phrases well, because it's easy to prove that all phrases appearing once in the document will get a score of one. As a consequence, it cannot distinguish proper keyword candidates from all low-frequency phrases. To solve this problem, we transform Accessor Variety(AV) into Document Frequency Accessor Variety(DF-AV).

We investigate how keywords are distributed across the document and discover that keywords are usually specialized words and words around keywords are usually common words. Document Frequency(DF) are usually used to distinguish specialized words and common words. It is generally admitted that words with high document frequency are usually common words. Therefore, if a phrase is surrounded by words with high document frequency, it's very likely to be a keyword candidate. This leads to DF-AV.

We calculate Document Frequency of words based on Chinese Gigaword corpus, which consists of around 1.4 million news articles. For a phrase string phr, define DF-AV and score of phr as follows:

$$DFAV_L = \sum_{w \in S_L} \log doc\_freq(w)$$

$$DFAV_R = \sum_{w \in S_R} \log doc\_freq(w)$$

$$Score(phr) = DFAV_L(phr) \times DFAV_R(phr)$$

A maximum length limits the number of words combining a phrase string. All phrases whose score higher than a threshold will be selected as keyword candidates. A low threshold will raise coverage rate of real keywords, but, on the other hand, it will result in more non-keyword phrases involved. Therefore, a proper threshold is needed to keep the balance.

3.2 TextRank on a Phrase Network

TextRank is a graph-based ranking algorithm inspired by famous PageRank [22]. TextRank transfers the document into a network of words, in which an edge between words stands for a relation between words. The importance of a word is determined by the importance of its neighbours. Formally, denote $G = (V, E)$ a directed graph with the set of vertices V and set of edges E. For a given vertex V_i, denote $In(V_i)$ the set of vertices with edges to V_i, $Out(V_i)$ the set of vertices with edges from V_i. The score of a vertex V_i is defined as follows:

$$S(V_i) = (1 - d) + d * \sum_{j \in In(V_i)} \frac{1}{|Out(V_i)|} S(V_j)$$

where d is a damping factor between 0 and 1. TextRank can be computed either iteratively or algebraically, like PageRank. In previous work [18, 20], a vertex of the graph stands for a single word. keywords are generated from combinations of top-ranked words. This method cannot ensure all generated keywords are independent linguistic entities. Additionally, not all words within a keyword can be ranked among top k. To improve this, we run TextRank on a phrase network, ranking phrases directly.

Based on keyword candidates selected by DF-AV, we build a graph with vertices standing for phrases. Usually, co-occurrence of words within a window of n determines a link between the words. We extend this relationship to words and phrases. Take word sequence "A B C D E" as an example. Each letter stands for a word. Suppose "BC", "CD", "BCD" are keyword candidates selected by DF-AV. We build a neighboring graph as Fig.1 . A directed edge from one vertex to another vertex means the latter one is next to the former one on the sequence.

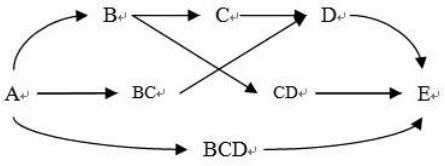

Fig. 1. Neighboring Graph

Based on neighboring graph, build phrase network graph according to window size n. To be specific, if there is a directed path no longer than n between two vertices, add a link between the vertices. Therefore, no linked vertices will share the same word. For instance, there are no links between "B", "BC", and "BCD". The phrase network graph based on Fig.1 with $n = 2$ is shown as Fig.2 .

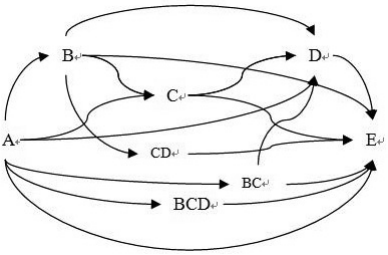

Fig. 2. Phrase Network Graph

Based on phrase network graph, we can compute importance of each vertex iteratively. Top-k ranked phrases are selected as keywords by the framework.

4 Improvement by Domain Knowledge

For scientific articles, choices of words vary between different domains. Especially, keywords are usually made up of specialized words, most of which are unique to the domain. Therefore, taking advantage of domain knowledge can improve performance of keyword extraction. In previous work [23, 24], thesaurus or Wikipedia are used as domain knowledge. They are usually of high quality but often not quite adaptive and construction of such resources is highly costly. However, through some online scientific article retrieval system, quantities of author-assigned keywords of a certain domain are available. Though some of those keywords are not quite normative, they can provide useful domain knowledge. In this section we'll show how to take advantage of raw known keywords to improve performance of keyword extraction.

4.1 Length of Keyword

There are many characteristics of keywords varying between domains. Length of keyword is a typical one. To show this characteristic, we choose four domains varying largely from each other, which are ethnology, petroleum, mathematics and international law. We do statistics of length of keywords from 1000 documents of each domain and show average length and distribution of length as Table.1 and Fig.3 respectively.

Table 1. Average Length of Keyword in Different Domains

Domain	ethnology	petroleum	mathematics	international law
Ave. Len.	3.54	4.16	4.62	4.48

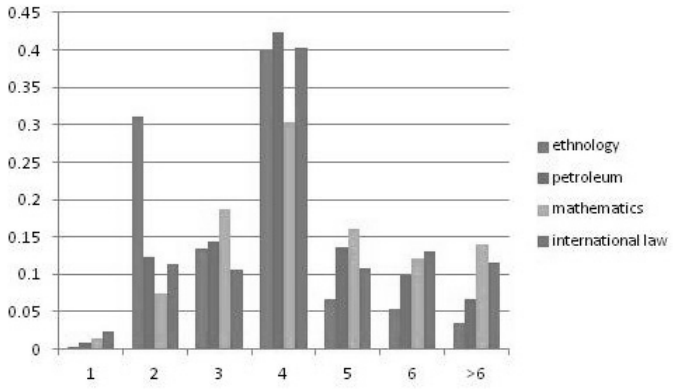

Fig. 3. Distribution of Length of Keyword

It's obvious length of keyword vary between domains. Keywords of ethnology tend to be short, while keywords of mathematics contain many longer ones. We will take advantage of this characteristic in two ways. First, we modify phrase-based TextRank graph into a weighted one. The new score is as follows.

$$WS(V_i) = (1-d) + d * \sum_{j \in In(V_i)} \frac{w_{ji}}{\sum_{V_k \in Out(V_i)} w_{jk}} WS(V_j)$$

We define $w_{ij} = 0.5 + 0.5 \times n_{len(V_j)} / \max(n_k)$, where n_k is the number of keywords with length k for this domain. Second, we use the same weight as a multiplied factor to TextRank score. In this way, we can eliminate keywords that are too long or too short.

4.2 Components of Keyword

We discover that components of keyword also vary largely between domains. We do statistics on words forming keywords and find that distribution of words are unique to the domain. For example, the word "random" is the most frequent word appearing in keywords of mathematics, while it never appears in a keyword of ethnology. And the word "culture" is most likely to be seen in a keyword of ethnology, but it only appear twice in keywords of mathematics. What's more, in the same domain, words have different possibility to act as starting word or ending word of keywords. For example, "system" and "equation" are among most

frequent ending words of keywords, but they never act as starting word unless they act as keywords independently. These phenomena are apparently useful for keyword extraction.

To employ such information into keyword extraction, we use it to eliminate irrelevant keyword candidates. If a keyword candidate starts with or ends with a word that never appears in this position, or it contains a word that never appears in the domain, we will discard this candidate. In order that out-of-vocabulary words will not be discarded, we require that related words must be common words with high document frequency.

4.3 High-frequency Keyword

Some of the keywords are frequently selected as keywords, especially those words indicating area of research or popular method. Thesaurus of the domain is a common resource for such specialized words. However, not every domain has such a thesaurus and it's costly to build a thesaurus. At this time, we can take advantage of quantities of author-assigned keywords. statistics show that about half the keywords are selected as keyword more than once in a certain domain and the most frequent keyword serves about 1/20 of all documents.

Based on intuition that high-frequency keywords are more likely to be keywords of other documents, we increase TextRank score of such keywords. We multiply the score by a weight $w_f = \sqrt[3]{freq(phr)}$, where $freq(phr)$ is the frequency of phr. The top-k ranked keywords according to the weighted score are selected as the extracted keywords.

5 Experiments and Evaluations

In this section, we first introduce the experimental settings in detail. Then we present the experimental results and give an analysis.

5.1 Experimental Setting

Dataset There are a few datasets for keyword extraction in English. However, similar datasets for Chinese are rare. So we retrieve our data from *cnki.net*. We choose four domains, ethnology, petroleum, mathematics and international law. For each domain, we retrieve title, abstract and author-assigned keywords of 100 randomly-selected documents as test set and author-assigned keywords of another 1000 randomly selected documents as domain knowledge. It is notable that we discard documents whose keywords never appear in the abstract, since our method is an extraction method from text, which determines unseen keywords cannot be dealt with. We take author-assigned keywords as standard keywords, even though some of the keywords might be inappropriate.

Pre-processing. We used a perceptron-based tool implemented based on [25] to do word segmentation on all titles, abstracts and keywords. And we do statistics of known keywords to obtain domain knowledge. When calculating length of

keyword, we treat single English letter and punctuation as length 1 and a whole English word as length 2, since 2-character words are most frequent in Chinese.

Evaluation. Following evaluation method of SemEval-2010 Task 5 [1], we show P,R,F1 of top 5, top 10 and top 15 ranked keywords, where F1 is the harmonic average of P and R. When averaging F1 across documents, we calculate macro-average and micro-average of F1 and take the mean of macro-average and micro-average as metrics of performance.

5.2 Experimental Results

To demonstrate difficulty of keyword extraction of different domains, we adopt the method of Term Frequency Inverted Document Frequency(TF-IDF) as a baseline system. It ranks all phrase strings using TF-IDF and choose top-k ranked phrases as keywords. When extracting keywords using our framework. For candidate selection, we set maximum word number as 4 and threshold as 100 to keep a balance of recall and precision.

Based on Phrase-based TextRank, we add domain knowledge one by one to show its effects on keywords extraction. Our experimental results are shown on Table 2, in which + means add this information based on the above system.

Table 2. Experimental Results of Keyword Extraction in Different Domains

	ethnology			petroleum			mathematics			international law		
	Top5	Top10	Top15	Top5	Top10	Top15	Top5	Top10	Top15	Top5	Top10	Top15
TF-IDF	0.243	0.234	0.201	0.108	0.141	0.148	0.115	0.122	0.127	0.211	0.189	0.166
TextRank	0.312	0.249	0.201	0.179	0.184	0.173	0.167	0.176	0.173	0.287	0.238	0.197
+component	0.319	0.253	0.199	0.176	0.184	0.176	0.170	0.176	0.176	0.285	0.239	0.196
+length	0.326	0.256	0.203	0.181	0.186	0.176	0.172	0.179	0.178	0.290	0.242	0.197
+high-freq	0.342	0.258	0.205	0.202	0.201	0.180	0.180	0.187	0.183	0.300	0.249	0.199

The results show the framework based on TextRank over a graph network can extract keywords effectively. There is a major improvement over TF-IDF, though it seems that improvement of top 15 is relatively small, which is because average keyword number is around 5, leading to precision lower than 35% even for the best case. Domain knowledge simply from known keywords can improve performance of keyword extraction, and the improvement is especially significant for Top 5 results. It's a simple and effective way to improve keyword extraction result from an unsupervised method. However, as domain knowledge added one by one, improvement might not be so significant, because targets of different domain knowledge work on might be overlapped. Among three aspects of information, improvement of high-frequency keywords is obvious, while improvement of components is not very stable, because the number of known keywords is limited and it is impossible to cover every possible keyword composition characteristics.

Comparing between domains, we can see that performance of ethnology and international law is much better than the other two domains. tf-idf results show

directly that keywords are easily to extract from ethnology and international law via frequency method. We analyse this phenomenon and find some objective reasons. First, words from those two domains are more similar to news so that precision of word segmentation is better, while the other domains vary from news largely. Second, documents from petroleum and mathematics contain many English words and symbols, and document structure is more complicated. It adds difficulty to keyword extraction.

Though introducing domain knowledge shows improvement to keyword extraction, general performance of keyword extraction is not ideal, especially for petroleum and mathematics. How to narrow the gap between domains and improve performance is our next task. What's more, though our proposed way to take advantage of domain knowledge is simple and effective, it relies on coverage and quality of known keywords. We will investigate how to combine unsupervised and supervised methods to build a better keyword extraction system.

6 Conclusion

This paper shows how to improve TextRank based framework for keyword extraction on Chinese scientific articles using domain knowledg. We first select keyword candidates by DF-AV. Then, based on selected candidates, we build a phrase network graph and run TextRank algorithm to select top-k ranked keywords. We use known keywords as domain knowledge to improve keyword extraction, with information of length of keyword, components of keywords and high-frequency keywords. Experimental results show that domain knowledge can improve performance of keyword extraction generally.

Acknowledgments. This research was partly supported by National Natural Science Foundation of China (No. 61370117, 61333018), Major National Social Science Fund of China (No. 12&ZD227), National High Technology Research and Development Program of China (863 Program) (No. 2012AA011101).

References

1. Kim, S.N., Medelyan, O., Kan, M.Y., Baldwin, T.: Semeval-2010 task 5: Automatic keyphrase extraction from scientific articles. In: Proceedings of the 5th International Workshop on Semantic Evaluation, pp. 21–26. Association for Computational Linguistics (2010)
2. Frank, E., Paynter, G.W., Witten, I.H., Gutwin, C., Nevill-Manning, C.G.: Domain-specific keyphrase extraction (1999)
3. Tomokiyo, T., Hurst, M.: A language model approach to keyphrase extraction. In: Proceedings of the ACL 2003 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment, vol. 18, pp. 33–40. Association for Computational Linguistics (2003)
4. Paukkeri, M.S., Nieminen, I.T., Pöllä, M., Honkela, T.: A language-independent approach to keyphrase extraction and evaluation. In: COLING (Posters), pp. 83–86 (2008)

5. Barker, K., Cornacchia, N.: Using noun phrase heads to extract document keyphrases. In: Hamilton, H.J. (ed.) Canadian AI 2000. LNCS (LNAI), vol. 1822, pp. 40–52. Springer, Heidelberg (2000)
6. Nguyen, T.D., Kan, M.-Y.: Keyphrase extraction in scientific publications. In: Goh, D.H.-L., Cao, T.H., Sølvberg, I.T., Rasmussen, E. (eds.) ICADL 2007. LNCS, vol. 4822, pp. 317–326. Springer, Heidelberg (2007)
7. Hulth, A.: Improved automatic keyword extraction given more linguistic knowledge. In: Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing, pp. 216–223. Association for Computational Linguistics (2003)
8. Hulth, A.: Combining machine learning and natural language processing for automatic keyword extraction. Department of Computer and Systems Sciences (Institutionen för Data-och systemvetenskap), Univ. (2004)
9. Witten, I.H., Paynter, G.W., Frank, E., Gutwin, C., Nevill-Manning, C.G.: Kea: Practical automatic keyphrase extraction. In: Proceedings of the Fourth ACM Conference on Digital Libraries, pp. 254–255. ACM (1999)
10. Liu, F., Pennell, D., Liu, F., Liu, Y.: Unsupervised approaches for automatic keyword extraction using meeting transcripts. In: Proceedings of human language technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp. 620–628. Association for Computational Linguistics (2009)
11. Matsuo, Y., Ishizuka, M.: Keyword extraction from a single document using word co-occurrence statistical information. International Journal on Artificial Intelligence Tools 13(01), 157–169 (2004)
12. Ercan, G.: Automated text summarization and keyphrase extraction. PhD thesis, bilkent university (2006)
13. Liu, Z., Li, P., Zheng, Y., Sun, M.: Clustering to find exemplar terms for keyphrase extraction. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, vol. 1, pp. 257–266. Association for Computational Linguistics (2009)
14. Krapivin, M., Autayeu, M., Marchese, M., Blanzieri, E., Segata, N.: Improving machine learning approaches for keyphrases extraction from scientific documents with natural language knowledge. In: Proceedings of the Joint JCDL/ICADL International Digital Libraries Conference, pp. 102–111 (2010)
15. Zhang, C.: Automatic keyword extraction from documents using conditional random fields. Journal of Computational Information Systems 4(3), 1169–1180 (2008)
16. Kim, S.N., Medelyan, O., Kan, M.Y., Baldwin, T.: Automatic keyphrase extraction from scientific articles. Language Resources and Evaluation 47(3), 723–742 (2013)
17. Lopez, P., Romary, L.: Humb: Automatic key term extraction from scientific articles in grobid. In: Proceedings of the 5th International Workshop on Semantic Evaluation, pp. 248–251. Association for Computational Linguistics (2010)
18. Mihalcea, R., Tarau, P.: Textrank: Bringing order into texts. Association for Computational Linguistics (2004)
19. Wan, X., Xiao, J.: Collabrank: Towards a collaborative approach to single-document keyphrase extraction. In: Proceedings of the 22nd International Conference on Computational Linguistics, vol. 1, pp. 969–976. Association for Computational Linguistics (2008)
20. Liu, Z., Huang, W., Zheng, Y., Sun, M.: Automatic keyphrase extraction via topic decomposition. In: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pp. 366–376. Association for Computational Linguistics (2010)

21. Feng, H., Chen, K., Deng, X., Zheng, W.: Accessor variety criteria for chinese word extraction. Computational Linguistics 30(1), 75–93 (2004)
22. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web (1999)
23. Hulth, A., Karlgren, J., Jonsson, A., Boström, H., Asker, L.: Automatic keyword extraction using domain knowledge. In: Gelbukh, A. (ed.) CICLing 2001. LNCS, vol. 2004, pp. 472–482. Springer, Heidelberg (2001)
24. Coursey, K.H., Mihalcea, R., Moen, W.E.: Automatic keyword extraction for learning object repositories. Proceedings of the American Society for Information Science and Technology 45(1), 1–10 (2008)
25. Zhang, Y., Clark, S.: Syntactic processing using the generalized perceptron and beam search. Computational Linguistics 37(1), 105–151 (2011)