Automatic Keyword Extraction for Scientific Literatures Using References

Yanchun Lu, Ruixuan Li, Kunmei Wen, Zhengding Lu
School of Computer Science and Technology
Huazhong University of Science and Technology
Wuhan, China
{yanchunlu, rxli, kmwen, zdlu}@ hust.edu.cn

***Abstract*—References provide some important clues for detecting keywords of the scientific literatures. We propose a unified framework based on word co-occurrence and topic distribution using references to extract top-k single keywords, and remove words within a range of topics. For those multiword keywords, we use LocalMaxs algorithm and apply the Co-occurrence Cohesion Degree to measure the "glue" of the n-gram. Experimental results show that our keyword extraction method by using references can obviously improve the performance of precision, recall and F-measure compared to other keyword extraction methods.**

Keywords—topic modelling; keyword extraction; topic distribution; scientific literature; reference

I. Introduction

Extracting keywords from scientific literatures can provide researchers with informative summarization words of the literatures. Although there are many tagging interfaces for researchers to tag the literatures manually, this behavior can be both subjective and labor intensive. In contrast, we would like to extract important words in an objective and automatic way. Many keyword extraction approaches consider either properties of words in the document collection or external resources like thesauri. In this paper, we incorporate the references of the literature to improve the performance of keyword extraction.

Keywords in a document are words that are both important in that document and discriminative within the whole collection. Previous keyword extraction methods include TF-IDF heuristics, graph-based method, and machine learning based methods. However, they do not consider the references of the literature, while keywords extraction can actually benefit from the consideration of the references. References are relevant to the literature so that the literature often share the same set of keywords with their references.

In addition, we take topic distributions of the literatures and words into consideration. A word that occurs in a number of documents on the same topic has more discriminative power than a word occurring in the same number of documents but occur in many different topics. The more similar the main topics of a word are to that of a literature, the more relevant it is to the literature.

Some academic terminologies consist of more than one word, and how to get these key phrases is still a challenge issue. LocalMaxs is one of such methods that extract multiword lexical units (MWUs) from documents. Motivated by its idea, we apply the Co-occurrence Cohesion Degree to measure the "glue" of the n-gram to extract the most likely MWUs. To overcome the low frequency of compound terms, we select a mediate compound terms distribution metric called Co-occurrence Cohesion Degree to identify the multiword by using references.

This paper proposes a method using academic references as ancillary resources to mine keywords of the literatures. We consider the co-occurrences of terms in the literature and reinforce the co-occurrences using its references to find a small set of words as candidate keywords. We use the topics of terms and literatures to refine the candidate keywords, and then apply topic entropy to filter out the candidate keywords distributed over a range of topics. Moreover, in order to find the phrase of keywords, Co-occurrence Cohesion Degree is applied into LocalMaxs.

II. Related Work

Extracting keywords from a text is closely related to ranking words with respect to their relevance to the text. TF-IDF heuristics (Spärck Jones, 2004) is one of such methods. TF-IDF assumes that words in documents are independent, while in real cases words are mutually correlated. Matsuo et al (2004) apply chi-square measure to determine the bias of word co-occurrences in the text which is then used to rank words and phrases.

Keyword extraction can be treated as a supervised machine learning problem (Frank et al, 1999). Xu et al (2010) introduce several novel word features for keyword extraction and headline generation. Somol et al (2006) present a framework that use traditional feature selection algorithms for building a

978-1-4799-6270-9/14/$31.00 ©2014 IEEE

subset of specified properties, including Sequential Forward Selection (SFS), Sequential Forward Floating Selection (SFFS) and Oscillating Search (OS). To extract keyword phrases, LocalMaxs algorithm isused to extract Multiword lexical units (MWUs) from documents (da Silva et al, 1999). Selection (SFFS) and Oscillating Search (OS). To extract keyword phrases, LocalMaxs algorithm is used to extract Multiword lexical units (MWUs) from documents (da Silva et al, 1999).

In this paper, we focus on the keyword extraction using literatures' references, which is largely ignored by previous works.

III. CO-OCCURRENCE AND TOPIC BASED KEYWORD EXTRACTION METHOD DESCRIPTION

The keyword extraction of our method is divided into four steps: 1) literature pre-processing; 2) co-occurrence graph construction; 3) topic distribution calculation; 4) key phrase identification.

A. Literature Pre-processing

In this step, we mark parts of speech and remove stop words. Specifically, we use TreeTagger to process the literatures and get part-of-speech tagger of each word. We remove some frequent words as stop words. There are many parts of speech tags for words in TreeTagger, such as JJ stands for Adjective, RB stands for Adverb, NN stands for Noun. In the previous research, it is shown that keywords are mostly noun or noun phrase. So we focus on the parts of speech tags start with "N", "V" and "J" in order to consider some noun words and phrases with modified with adjective.

B. Co-occurrence Graph Construction

After pre-processing the literatures, we obtain the distinct term set $W_l = (w_1, w_2, ..., w_N)$ and count the frequency of each term to form a term frequency vector $L = (tf_1, tf_2, ..., tf_N)$. We separate each reference by period or comma to get sentences of the references, and then count the co-occurrence times of each term pair $\{(w_i, w_j): w_i, w_j \in W_l, w_i \neq w_j\}$ in these sentences and literatures. Traverse all the pairs, repeat counting the co-occurrence of each pair to get the co-occurrence graph of the terms in the literature so as to consider its centrality and also strengthen it using the occurrence of the term pairs in references.

Since we get the co-occurrence graph of the terms, there is a term co-occurrence vector for each term w_i, which is a co-occurrence distribution over the other terms. We use Jensen-Shannon (JS) divergence between co-occurrence distribution of each term and that of the literature l to evaluate the relevance of term w_i to l.

$$\text{Term-relecance}(l, w)$$

$$= \frac{1}{2} D_{KL}(\text{CoDis}(l) \| m) + \frac{1}{2} D_{KL}(\text{CoDis}(w) \| m) \quad (1)$$

Where $m = \frac{1}{2}$ (CoDis(l)+CoDis(w)), CoDis(t) is the co-occurrence distribution vector, and Term-relecance(l, w) is the relevance of term w to literature l. The smaller the Term-relecance value, the more relevant term w is to literature l.

C. Topic Distribution Calculation

Topics are the latent semantics of the documents, which can be inferred by topic models. The topic distribution of literature l is denoted as topics(l), the topic distribution of term w is denoted as topics(w). The cosine similarity and Jensen-Shannon (JS) divergence between the topic distributions of literature and term are denoted by Topic_Sim_Cos(l, w) and Topic_Sim_Jsd(l, w) respectively. Using the two different similarity metrics, we can get two sets of sorted candidate keyword sets.

We select top probability topics as the main topics for literatures and terms. The number of main topics is empirically selected as 2, 3 or 4 in our dataset. The main topics of literature l is denoted by Main_topics(l), and that of term $\mathbf{w}$ is denoted by Main_topics(w). The intersection size between them is denoted as Same_Topic_Count(l, w). After counting the Same_Topic_Count between literature l and each term w, we can compare the count between literature l and each term w, and then get two sets of sorted candidate keyword sets.

Furthermore, we calculate the entropy of the literature's topic distribution as Equation (2).

$$f_{topic\text{-}entropy}(w) = -\sum_{t=1}^{K} prob_{ti} \log(prob_{ti}) \quad (2)$$

Where $prob_{ti}$ is the probability of topic t_i for term w. We use this metric to filter the words within a range of topics.

D. Key Phrases Identification

For the metrics mentioned above, we can use them to identify the single keywords, there are many keywords in academic literatures are multi words, in order to capture these multi words. We propose a method called Co-occurrence Cohesion Degree (CoCD) to identify the MWUs of the literature so as to identify key phases. The main idea of CoCD is: 1) listing all combination of 2 or 3 or 4-gram continuous term groups in each segment split by comma, period, colon or question mark; 2) count the occurrences of each term groups, and each single term in them; 3) calculation of CoCD for each term groups; 4) using *LocalMaxs* algorithm to select key phrases.

The value of CoCD is affected by two factors. One is the document frequency, the higher the better; and the other is the co-occurrences compared to the occurrences of the single term. We formulate the two factors by using the co-occurrences of the N-gram term groups divided by the average occurrence of each single term, multiplied by the document frequency, which is shown in Equation (3).

$$\text{CoCD}(w_i, w_{i+1}, ..., w_{i+k}) =$$

$$\frac{Co\_occurrences(w_i, w_{i+1}, ..., w_{i+k}) \times DF(w_i, w_{i+1}, ..., w_{i+k})}{\sum_{i}^{i+k} occurences(w_i) / (k+1)} \quad (3)$$

Where $1 \le k \le 3$, *COoccurrences* is the occurrence of the continuous terms $(w_i, w_{i+1}, ..., w_{i+k})$, DF$(w_i, w_{i+1}, ..., w_{i+k})$ is the number of literatures they occur, occurrences(w_i) is the occurrences of term w_i. This formula quantifies the cohesion degree which can be used in the *LocalMaxs* algorithm as the metric to identify MWUs. A MWU is a phrase with the greatest local probability. *LocalMaxs* algorithm elects each n-gram whose cohesion value is greater than the others.

After the previous four-step processing, assume we set the number of keywords to *K*. Firstly, we can get the top *K* candidate single keywords after co-occurrence distribution and co-occurrence distribution respectively, denoted by *A* and *B* respectively. Secondly, we use topic entropy metric to filter out words within a wide range of topics, whose threshold is decided by the average topic entropy value. The words with higher value than the average value form the word set *C* to be removed. Thirdly, we use the words in $D=A\cap B-C$ as the candidate single keywords*,* which are those words in both *A* and *B* but not in *C*. Fourthly, we use *LocalMaxs* algorithm to extract all the multiword keywords candidate noted by *E*, and then select the multiword keywords whose term are in *D*, while remove these single candidate keywords from *D*. Finally, we get the extracted keywords including single and multiword keywords, whose number is at most *K*.

IV. Experimental Evaluation

A. Dataset

The dataset we use is downloaded from ACM Digital Library, which consists of 10943 PDF documents. For each of the articles, keywords are available in the content of the PDF file. We use 1000 of the literatures in our experiments, which have 10418 references with abstracts. There are 7865 distinct literatures and 4287 distinct keywords. Each article has 1 to 10 keywords, with an average of 4.5 words. We consider this set of keyword annotations as a golden standard and evaluate extracted keywords by computing precision and recall for this data set.

B. Experimental Results

We test 10 approaches for keyword extraction as shown in Table 1. In the experiments, we test three metrics for topic similarity, which are using Cosine, Jensen-Shannon (JS) divergence and the same topic number as the topic similarity metric respectively, and combined with the co- occurrence distribution with references respectively. Moreover, when considering the topic distribution, we evaluate the word topic distribution, using topic entropy to filter out the words within a wide range of topics. We use recall, precision and F1 measure as the metrics. The standard keywords are the keywords of the literature itself, they are provided by the authors. We test each method to extract top *N* keywords and compare them with the origin keywords of the literatures.

We calculate the precision, recall and F-Measure with different number of the keywords, and compare the performance of the ten methods mentioned above, which is shown in Figure 1. Table 2 gives the exact numbers for precision and recall for the top 3 keywords set for all the methods. The number of keywords *K* is set to 3 because we analysis form the experiments, when *K* is higher than 5, the precision will decrease sharply. As we can see that the Co_To_Jsd_Refs and Co_To_Cos_Refs is much better than the others, with precision 0.458 and 0.438 respectively, recall 0.198 and 0.188 respectively. The Co_Topic_Cos_Res method is not so better than Co_Topic_Jsd_Res method due to the metric for topic similarity, where JS is more appropriate for topic distribution to measure topic similarity, while the cosine metric is usually used for vector spaces. The Co_Topic_Stn and Co_Topic_Stn_Res methods are the worst ones, for the reason that the same topic number is not so good when the topics of words is decided by the context, but the topics of the literature is decided by the whole words in the literature.

Overall, the performance of the method with references is better than without using references. Compared to TF method, the method considering references and topics are better.

TABLE I. Evaluation Methods in the Experiments

No	Item	Description
1	TF	Traditional TF methods for single literature
2	TF _Refs	Traditional TF methods for single literature with references
3	Co_Dis	JS divergence of Co-occurrence distribution for single literature
4	Co_Dis_Refs	JS divergence of Co-occurrence distribution for single literature with references
5	Co_To_Cos	Our method using JS divergence of Co-occurrence distribution, Cosine metric for topics and Topic Entropy to filter out words with scattered topics for single literature
6	Co_To_Cos_Refs	Our method using JS divergence of Co-occurrence distribution, Cosine metric for topics and Topic Entropy to filter out words with scattered topics for single literature with references
7	Co_To_Jsd	Our method using JS divergence of Co-occurrence distribution, JS divergence metric and Topic Entropy for topics for single literature
8	Co_To_Jsd_Refs	Our method using JS divergence of Co-occurrence distribution, JS divergence metric for topics and Topic Entropy to filter out words with scattered topics for single literature with references
9	Co_To_Stn	Our method using JS divergence of Co-occurrence distribution, Same topics number metric and Topic Entropy for topics for single literature
10	Co_To_Stn_Refs	Our method using JS divergence of Co-occurrence distribution, Same topics number metric for topics and Topic Entropy to filter out words with scattered topics for single literature with references

TABLE II. PRECISION, RECALL AND F1-MEASURE FOR 10 METHODS ON ACM PORTAL DIGITAL DATA

No	Test methods	Precisi-on	Recall	F1
1	TF	0.379	0.178	0.242
2	TF _Refs	0.403	0.174	0.243
3	Co_Dis	0.411	0.186	0.256
4	Co_Dis_Refs	0.421	0.189	0.261
5	Co_To_Cos	0.428	0.176	0.261
6	Co_To_Cos_Refs	0.438	0.188	0.263
7	Co_To_Jsd	0.428	0.179	0.261
8	Co_To_Jsd_Refs	0.458	0.198	0.276
9	Co_To_Stn	0.332	0.129	0.186
10	Co_To_Stn_Refs	0.361	0.135	0.196

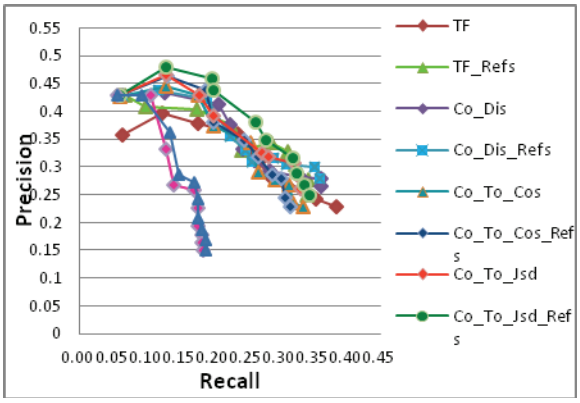

Fig. 1. Precision and recall for 10 methods

V. CONCLUSION

In this paper we investigate co-occurrence and topic based keywords extraction using references. We propose a four-step approach for solving this problem. We use the co-occurrence distribution of words to measure the relevance of each word in the literature, and use the topic model to automatically extract topics of the literatures and words, and perform similarity comparison between the literature's and word's topics to select the most relevant words. We also use topic entropy to filter out words within a wide range of topics. The co-occurrence of terms can be boosted by using references rather than only the single literature or the whole corpus, which is much more accurate and effective for keywords extraction. Finally, we use the *LocalMaxs* algorithm to identify multiword keywords without threshold, and then integrate the top-k keywords include single and multiword as keywords. When applying the approach on keywords extraction on scientific literatures, we experimentally validate the effectiveness of our method.

Acknowledgment

This work is supported by National Natural Science Foundation of China under Grants 61300222, 61173170 and 60873225, National High Technology Research and Development Program of China under Grant 2007AA01Z403, and Innovation Fund of Huazhong University of Science and Technology under Grants 2012TS052, 2012TS053 and 2013QN120. We sincerely thank the anonymous reviewers for their comprehensive and constructive comments.

References

[1] E. Frank, G. W. Paynter, I. H. Witten, C. Gutwin, and C. G. Nevill-Manning, "Domain-specific keyphrase extraction", In: *IJCAI*, 1999, pp 668–673

[2] J. da Silva, G. Dias, S. Guilloré, and J. Pereira Lopes, "Using LocalMaxs Algorithm for the Extraction of Contiguous and Non-contiguous Multiword Lexical Units", In: *Progress in Artificial Intelligence*, 1999, pp 113-132

[3] K. Spärck Jones, "A statistical interpretation of term specificity and its application in retrieval", *Journal of documentation*, Vol. 60, No. 5, 2004, pp 493–502

[4] P. Somol and P. Pudil, "Multi-Subset Selection for Keyword Extraction and Other Prototype Search Tasks Using Feature Selection Algorithms", In: *ICPR*, 2006, pp 736–739

[5] S. Xu, S. Yang and F.C.M Lau, "Keyword Extraction and Headline Generation Using Novel Word Features", In: *AAAI*, 2010, pp 1461–1466

[6] 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, 2004, pp 157–170