ISSN 2347 - 3983
Volume 8. No. 9, September 2020
International Journal of Emerging Trends in Engineering Research
Available Online at http://www.warse.org/IJETER/static/pdf/file/ijeter18892020.pdf
https://doi.org/10.30534/ijeter/2020/18892020

Key Phrase Extraction by Term Clustering using Proposed Graph Based Ranking Model Method

Kannan G[1], R. Nagarajan[2]
[1]Government Arts and Science College, Mayiladuthurai, India, kannanaucse@gmail.com
[2]Department of Computer and Information Science, Annamalai University, India,

ABSTRACT

This paper proposes a novel keyphrase extraction procedure which is referred as term clustering based keyphrase extraction (TCKE). The proposed graph-based keyphrase extraction method using TCKE is employed and the objective of this work is to extract the most important terms from the journal document. The proposed model consists of three sets namely document pre-processing, display of terms in graphical formats, and efficient retrieval of keywords. In pre-processing stage, the document is tokenized and removed the stop words from the text document. Further, the important terms are extracted and clustered into keyphrases. The term clustering proposed in this paper is employed to find the candidate keyphrases. Using this candidate keyphrases, keyphrase graph is constructed. The experimental analysis is carried out in journal article and the results depict that the proposed graph-based keyphrase extraction technique can effectively extracts the most important terms from journal document.

Key words: Graph-based approach, Keyphrase extraction, Term clustering, Tokenization.

1. INTRODUCTION

The task of extracting keywords from the text document or webpage is known as keyphrase extraction [1]. Also, a keyphrase extraction process is used to get information on the basis of a given query from the vast database. Many real world keyword extraction applications are available, for example, web search, text summarization, twitter trends, etc.

Because of the fast increasing use of the internet, more than enough user-generated websites and applications have been created that are coming daily. Most of the websites and applications are short text, such as twitter, quora, and stack overflow, etc. The user-generated content is becoming increasingly fragmented and shorter. The managing of a large number of short user-generated content has become more and more important for the web application service provider [2] [3]. Their management is based on the retrieval and accuracy of the information. Many researchers have recently analyzed the extraction of keyphrase from short text documents, and it is called as micro blogs. Recently, micro blogs attracted people to speak out and to communicate with others. There are so many sites for micro blogs but twitter is one of the most popular sites for micro blogs. A lot of works have been done in recent text analytics research and the rate of work in the future is also increasing rapidly.

There are numerous information retrieval techniques available. One of these is the manual extraction of keyphrases where the manual level reading and extraction of keywords for a document can be done. However, this is time-consuming and costly in terms of human resource utilization. The other method is to automatically extract keyphrases where documents are interpreted using computers. Representing a bunch of sentences using a few words is a very challenging task when it comes to the automation process. On the other hand, keyphrase extraction is currently one of the most well-known techniques in the field of research. Keyphrase extraction plays a key role in different fields such as text retrieval, text grouping, text resuming, and indifferent information processing fields.

The aim of this work is to extract the most important terms from the document, then the terms are clustered into keyphrases, and finally the extracted keyphrases are represented as keyphrase graph. Thus in this paper, the graph-based keyphrase extraction from documents is proposed and the keyphrase extraction procedure is referred as term clustering based keyphrase extraction (TCKE). The process of concatenation of two candidate keywords is called as term clustering. In this paper, the term clustering is used to find the candidate keyphrases. Using this candidate keyphrases, keyphrase graph is constructed.

The reminder of this paper is organized as follows: Section 2 deals with the literature survey denoting the recent works carried out on keyword extraction. Section 3 describes the proposed model which includes the representation of

documents and the model that we preceded and in section 4 deals with the experimental analysis. Finally, section 5 concludes the paper.

2. RELATED WORK

The rapidly increasing web-based culture adds an avalanche of data over the internet. The amount of growing unstructured and vast data on the Internet makes people more comfortable to read, yet renders it very difficult to access information and obtain knowledge from it. Keyphrase extraction is a trendy subject in the field of text processing, and there are several methods to extract keyphrases [1]. The search engine works on the basis of keyword extraction, such as entering a keyphrase in the search bar and displaying further suggestions in the search bar. After searching, google shows a number of links related to that keyphrase.

Wen et al., discussed the various classification methods of keyword extraction [4]. In their work, they performed classifiers to extract keywords from the news articles. They developed the candidate keyword graph on the basis of TextRank by calculating the similarity between the words as the nodes' transition probability; then they estimated the score of words by an iterative method, and finally picked the top N keywords as the final keywords.

To extend the work of keyword extraction from the graph-based models [4], many other researchers dragged this area in different aspects of the graph. Cao et al., described a method to enhance the graph-based keyword extraction, where they proposed an approach to calculate the importance of co-occurrence words in the documents and modeled it as a graph to identify more relative keyphrases [5]. Also, they introduced the word co-relation degree in the documents to enhance the performance while extracting the average number of keywords from the documents.

Islam et al. proposed a novel enhanced technique for extraction of keywords employing a random walk model by taking into account of the position of terms within the documents and terms' information gain corresponds to the entire document sets [6]. Moreover, they incorporated terms' mutual information with the help of the random walk model to extract keyword from the document. They created the random walk model by the TextRank. Previously, there are several types of random walk that have been developed and successfully applied in various applications, such as citation analysis, social networks, web link analysis, and so on. Their work of the post-processing stage is similar to that of Mihalcea et al., [7].

Wang et al., proposed a keyword extraction technique on the basis of WordNet and PageRank [8]. In this approach, initially the candidate word is denoted as an undirected graph with information related to nodes and relations of nodes as links. They have applied PageRank in the undirected graph to do word sense disambiguation (WSD). Then they pruned the word graph, and applied PageRank again on the resultant graph for keyword extraction. The outcomes obtained from their research depicted that the keyword extraction technique based on WordNet and PageRank is more efficient and practically applicable.

Li et al., authors proposed the graph-based ranking method by utilizing wikipedia as external information for extraction of short text keyword [9]. They introduced the wikipedia to better the quality of the short text content in order to address the shortage of poor knowledge in the short text content. The comprehensive studies done in their work showed that the graph-based ranking approach can better the F-measure, recall, and precision indices. They also mentioned that the TextRank graph-based ranking method is more appropriate for keywords extraction of short text content by exploiting the information available in wikipedia.

Mihalcea et al., described two innovative unsupervised approaches to extract keyword and sentence from the documents [10]. They converted the text in the form of the graph as structured data. They introduced the TextRank graph-based ranking models to process the text in the documents and showed how the approach can be favorably employed in the natural language text. The obtained results from TextRank graph-based ranking models have been compared successfully with existing published results.

Zhou et al., described an approach for ranking the result of keywords over the structured data [11]. Their work is based upon the schema graph-based method for keyword search that comprises a candidate network generator and its evaluation stage. By using this idea, the ranking process has been done to the keywords which result in optimized words from the document.

Litvak et al., implemented two modern methods, unsupervised and supervised graph-based syntactic methods, to identify the keywords from the text document [12]. Also, they have compared both of these methods. In the unsupervised method, they have used HITS algorithm for keyword extraction from the text and web documents. While in the supervised method, conventional space vector model has been employed to extract the keywords from the text document. The space vector model has been trained using the keyword in the summarized collection of documents.

According to Ohsawa et al., keyword extraction techniques are roughly divided into different approaches such as statistical approaches, machine learning approaches,

linguistic approaches, and other approaches [13]. Further authors discuss the keyword extraction on graph-based data and also they classified the various graph types. Most of the analysis is done on the co-occurrence graph because it is easy to compute and construct with the two words simultaneously. Ohsawa et al., proposed KeyGraph algorithm to extract the keyword by describing the stated important points in the documents, without depending on extra sources such as natural language processing tools or the large corpus of documents [13]. The presented algorithm is focused on the graph segmentation and presenting the co-occurrence between words in the documents into word clusters. Despite the fact that presented algorithm not utilizes the mean frequency of terms in the document corpus, the outcomes obtained from their experiment showed that the extracted keywords are more accurate.

Wang et al., described the sentiment analysis in twitter based on a novel graph model and the analyses data are crawled from twitter [14]. They have employed the literal meaning of hashtags as semi-supervised information in boosting classification setting to improve the performance of the presented graph model.

Brin et al., presented google a large scale exploration engine that makes extensive application of the framework available in the hypertext [15]. Google is modeled to crawl and index the web effectively so that the search results are more satisfied than the already available methods. They addressed the question of how to develop a practically large scale network that can take the advantage of the information additionally available in the hypertext.

3. TERM CLUSTERING BASED KEYPHRASE EXTRACTION (TCKE)

The proposed graph-based model consists of three separate steps for the efficient extraction of keywords from a given text. The proposed model requires document pre-processing, display of terms in graphical formats, and efficient retrieval of keywords. The steps are described in more depth in this module.

3.1 Step 1: Pre-Processing

Tokenization

Each word in the text document is considered as a terms called *t*. A collection of terms in a single document is defined as,

$$D = \{t_1, t_2, ..., t_{nt}\} \quad (1)$$

where *nt* denotes the total term counts in the text document and *nt* varies from document to document.

Stop Words Removal

A stop word dictionary is employed to remove the words such as ‘is’, ‘are’, ‘was’, ‘were’, ‘he’, ‘she’, ‘they’, ‘it’, ‘who’, ‘what’, ‘where’, when’, and so on from the document *T* [10].

Removal of Trivial Token

There are also certain terms in the collection of tokens despite the elimination of stop terms that cannot be a keyword that retains the potential to be a keyword to describe the whole text. It is also important to define and delete those terms. The method used to extract such terms can be described as follows.

Consider a term denoted as *w* that occurred in set *D* more than once. The complete measure of the frequency of the term *w* can be calculated as,

$$TF_w = \sum_{i=1}^{nr} (w == t_i) \quad (2)$$

where *nr* is the number of terms after removal of stop words in set *D*. If the *WF* of a term is less than the specified threshold value, then the corresponding term is removed from the set *D*.

3.2 Step 2: Construction of Graph using Term Clustering

Every term after the pre-processing stage is considered as node. In the representation of terms in the form of graph, each single word (term) is represented as node *V* and the relationship between the terms are represented as links *E*. The term graph is constructed with nodes and links as,

$$G = (V, E) \quad (3)$$

The resultant term graph is representation of all unique candidate words in graphical form.

Term Clustering

Based on the link between the two terms, the concatenation of two terms in both directions is carried out to form a concatenated phrase with two terms. This process is called as term clustering. Once the term clustering is done for all possible nodes, the candidate keyphrases set *K* is constructed. Using this candidate keyphrases, keyphrase graph is constructed.

The complete measure of the frequency of the keyphrase *k* can be calculated as,

$$KF_k = \sum_{i=1}^{n} (k == K_i) \quad (4)$$

where *n* is the number of keyphrases in set *K*.

3.3 Step 3: Keyphrase Extraction

Based on keyphrase graph and the value of *KF* of every

keyphrase, each document can be easily represented using the extracted keywords.

4. RESULTS AND DISCUSSION

The extraction of keyphrases can only be evaluated based on the accuracy that helps to predict or classify the given documents into appropriate classes. For the evaluation of the proposed graph-based model, the keyphrase extraction has been implemented in MATLAB with the system specification of Intel Core i7 processor, 500 GB hard disk and 8 GB RAM.

In this study, the document considered for analysis is the journal paper entitled optimal placement of distribution generation in micro grid using eagle strategy with particle swarm [16]. As per the procedures given in the section 3, the keyword graph is constructed using the proposed graph-based technique. Figure 1 shows the candidate keyword graph for top 29 keywords which are extracted using the proposed TCKE model. Table 1 shows the statistical indices and centrality measures. The candidate keyphrase graph is shown in Figure 2.

From Figure 2, it can be seen that the most important terms in the document taken for analysis are generation placement, micro grid, optimal placement, eagle strategy and particle swarm. By referring the actual keywords in the document from the journal website [16], the proposed TCKE model extracts almost same keyphrases from the document. Thus the proposed TCKE model is best suits for extraction of keyphrases from the text documents.

Table 1: Statistical indices and centrality measures

Nodes	Keywords	TF	Betweenness	Degree	Closeness
1	Optimal	22	32.11821789	0.25	0.01754386
2	Placement	16	9.024675325	0.142857143	0.016949153
3	Generation	19	4.74491342	0.142857143	0.014925373
4	Micro	16	1.376190476	0.107142857	0.014285714
5	Grid	15	6.478571429	0.142857143	0.016129032
6	Eagle	15	18.87424242	0.214285714	0.016949153
7	Strategy	18	10.43571429	0.178571429	0.016129032
8	Particle	28	12.58658009	0.142857143	0.014285714
9	Swarm	16	1	0.071428571	0.012048193
10	Technique	23	6.998015873	0.142857143	0.016666667
11	Optimization	26	18.41428571	0.178571429	0.015384615
12	Algorithm	41	38.34588745	0.321428571	0.020408163
13	Global	21	47.73163781	0.392857143	0.019607843
14	Exploration	18	6.138924964	0.178571429	0.016666667
15	Local	24	3.991666667	0.178571429	0.015151515
16	Real	17	3.050865801	0.178571429	0.015873016
17	Power	61	23.03892496	0.285714286	0.018518519
18	System	53	16.13160173	0.25	0.016949153
19	Voltage	23	1.042857143	0.107142857	0.013888889
20	Loss	38	3.050865801	0.142857143	0.014925373
21	Method	21	11.21165224	0.214285714	0.018518519
22	Bus	52	72.49924242	0.392857143	0.020833333
23	DG	52	52.00238095	0.357142857	0.020833333
24	Position	18	6.852453102	0.178571429	0.016129032
25	PSO	51	36.94336219	0.321428571	0.020833333
26	Search	16	2.53023088	0.142857143	0.015873016
27	Solution	22	10.57316017	0.142857143	0.017241379
28	ES	45	15.63863636	0.285714286	0.019230769
29	Load	29	1.174242424	0.071428571	0.013888889
30	load	29	2.705988456	0.103448276	0.014925373

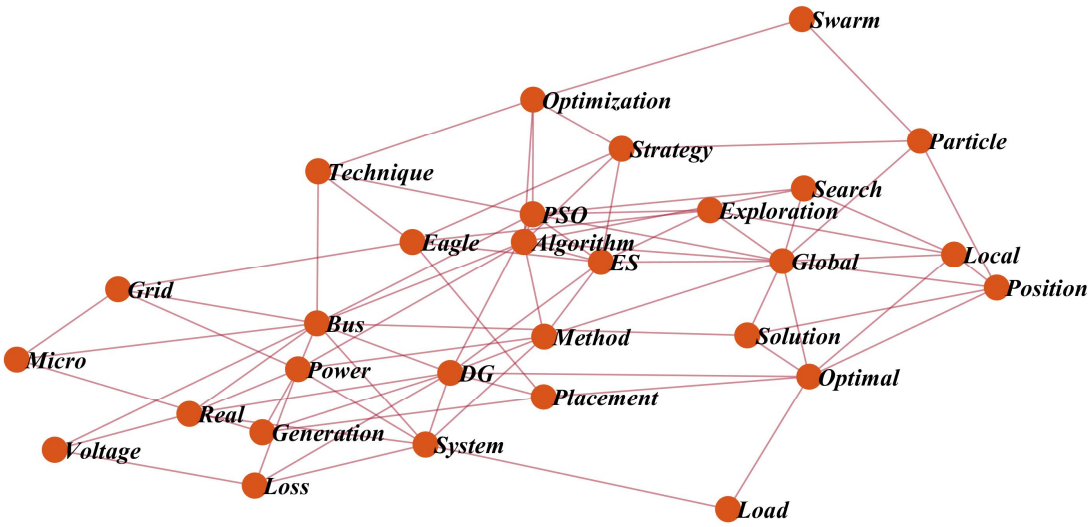

Figure 1: Keyword graph for top 29 keywords

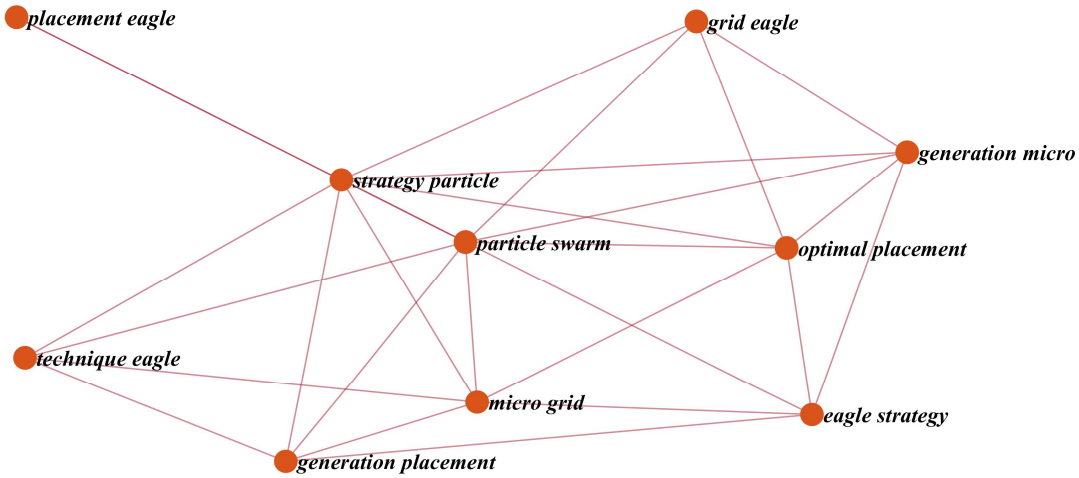

Figure 2: Keyphrase graph constructed using proposed TCKE

5. CONCLUSION

In this paper, a novel keyphrase extraction procedure has been proposed to construct the keyphrase graph; and extract the most import terms from the journal document. The proposed graph-based keyphrase extraction method using TCKE has been employed to extract the most important terms from the journal document. The experimental analysis has been carried out in journal article and the outcomes depicted that the presented TCKE keyphrase extraction technique can effectively extract the most important terms from journal document. The proposed TCKE model extracts almost same keyphrases from the document while referring the actual keywords from the journal website. Hence, the proposed TCKE model can be best suits for extraction of keyphrases from the various text or web documents.

REFERENCES

1. R. Nagarajan, S. Nair, P. Aruna, and N. Puviarasan. **Keyword Extraction using Graph Based Approach**, *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 10, pp. 25-29, 2016.
2. K.R. Shibu, and R. Suji Pramila. **A Novel Secret Key Generation Scheme for MANETs using Traffic Load to Avoid Active Attackers,** *International Journal of Emerging Trends in Engineering Research*, Vol. 8(5), pp. 1539-1544, 2020.
3. A.S. Girsang, F.H. Usman, and R.M. Sunarto. **Clustering Hostels Data for Customer Preferences using K-Prototype Algorithm,** *International Journal of Emerging Trends in Engineering Research*, Vol. 8(6), pp. 2650-2653, 2020.
4. Y. Wen, H. Yuan, and P. Zhang. **Research on Keyword Extraction Based on Word2Vec Weighted TextRank,**

2nd IEEE International Conference on Computer and Communications (ICCC), IEEE, 2016, pp. 2109-2113.

5. J. Cao, Z. Jiang, M. Huang, and K. Wang. **A Way to Improve Graph-Based Keyword Extraction,** *IEEE International Conference on Computer and Communications (ICCC), IEEE,* 2015, pp. 166-170.
6. M. R. Islam and M. R. Islam. **An Improved Keyword Extraction Method using Graph Based Random Walk Model,** *11th International Conference on Computer and Information Technology, IEEE*, 2008, pp. 225-229.
7. R. Mihalcea. **Graph-Based Ranking Algorithms for Sentence Extraction, Applied to Text Summarization**, *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, 2004, pp. 170-173.
8. J. Wang, J. Liu, and C. Wang. **Keyword Extraction Based on Pagerank**, *PacificPacific-Asia Conference on Knowledge Discovery and Data Mining, Springer*, 2007, pp. 857–864.
9. W. Li and J. Zhao. **Textrank Algorithm by Exploiting Wikipedia for Short Text Keywords Extraction,** *3rd International Conference on Information Science and Control Engineering (ICISCE)*, 2016, pp. 683–686.
10. R. Mihalcea and P. Tarau. **Textrank: Bringing Order into Text,** *Proceedings of the 2004 conference on empirical methods in natural language processing*, 2004, pp. 404–411.
11. J. Zhou, X. Yu, Y. Liu, and Z. Yu. **Ranking Keyword Search Results with Query Logs,** *IEEE International Congress on Big Data, IEEE*, 2014, pp. 770–771.
12. M. Litvak and M. Last, **Graph-Based Keyword Extraction for Singledocument Summarization,** *Proceedings of the workshop on Multisource Multilingual Information Extraction and Summarization Summarization. Association for Computational Linguistics*, 2008, pp. 17–24.
13. Y. Ohsawa, N. E. Benson, and M. Yachida. **Keygraph: Automatic Indexing by Co-occurrence Graph Based on Building Construction Metaphor,** *Proceedings IEEE International Forum on Research and Technology Advances in Digital Libraries, IEEE*, 1998, pp. 12–18.
14. X. Wang, F. Wei, X. Liu, M. Zhou, and M. Zhang. **Topic Sentiment Analysis in Twitter: a Graph Graph-Based Hashtag Sentiment Classification Approach,** *Proceedings of the 20th ACM international conference on Information and knowledge management, ACM*, 2011, pp. 1031–1040.
15. S. Brin and L. Page. **The Anatomy of a Large Large-Scale Hypertextual Web Search Engine,** *Computer networks and ISDN systems*, vol. 30, pp. 107–117, 1998.
16. K. Santhosh, and R. Neela. **Optimal Placement of Distribution Generation in Micro-Grid using Eagle Strategy with Particle Swarm Optimizer**, *International Journal of Pure and Applied Mathematics*, Vol. 118(18), pp. 3819-3825, 2018.