Proceedings of the Second International Conference on Intelligent Computing and Control Systems (ICICCS 2018)
IEEE Xplore Compliant Part Number: CFP18K74-ART; ISBN:978-1-5386-2842-3

Comparative Study on Keyword Extraction Algorithms for Single Extractive Document

Shweta Ganiger
School of Computer Science & Engineering
KLE Technological University
Hubbli, Karnataka, India
Shweta.b.ganiger@gmail.com

K.M.M.Rajashekharaiah
School of Computer Science & Engineering
KLE Technological University,
Hubbli, Karnataka, India
kmmr@bvb.edu

***Abstract*— The Automatic Text Summarization is most discussed area in Text Mining; there are various techniques available in text mining for text summarization. The two type of summarization are the extractive and abstractive text summarization. The main aim of text summarization is to obtain the concise meaningful text from the original text document. Keywords plays an important role in building a summarization text, there are several keyword extraction algorithms were proposed. In this paper, we implemented most popular keyword extraction algorithms the TF-IDF(a baseline algorithm), TextRank and RAKE algorithm. These keywords extraction algorithms were tested their effectiveness in finding important keywords from single document; the retrieved keywords are compared with the manually selected keywords. The comparison is performed to check the performance of each implemented algorithms with each other and with manually selected keywords.**

Keywords- Automatic Text Summarization, Keywords extraction, baseline, TF-IDF, TextRank, RAKE.

I. INTRODUCTION

Text Summarization is the area of Text Mining using natural language processing; Text mining is used for retrieving the useful amount of data from large dataset. As the data from internet source is overflowing in webs, the overload of the data increases day by day the user cannot find the important data from large data so the automatic text summarization is gaining its importance.

The text summarization has become significant from the early sixties but the need was different in those days, as the storage capacity was very limited, summaries were needed to be stored instead of large documents. Those days the storage of data was manually performed, nowadays the data availability growth is increasing day-by-day, the need for converting such data into useful information and knowledge is required. When a user enters a query into the system, the response which obtained consists of several web pages with plenty of information and the user is unable to receive the important knowledge. The research in this field is gaining the attention of researchers in today's fast growing age due to the exponential growth in the quantity and complexity of information sources on the internet. The information and knowledge gained can be used for applications ranging from business management, production control, and market analysis, to engineering design and science exploration. A summary is a text that is produced from one or more documents, which contains a fundamental portion of the information from the original document, and that is no longer than even half of the original text.

The text summarization can be classified into extractive summarization and abstractive summarization[6]. The extractive summarization extracts the same important keywords, key phrases, sentences present in the original text document whereas the abstractive summarization involves in creation of different keywords or sentence that highlights the main idea in original document. The most of time extractive summarization is carried out for summarization as it gives better result compared to the abstractive summarization [1]. The text summarization technique uses the keyword as basic building block of summary, by extracting important keywords the summarization is obtained from large set of text document. Keyword extraction has become a difficult task in text mining, as performance of keyword application majorly depends on speed and quality of keyword extracted. There are various keyword extraction algorithms proposed which gives different output for same document[3]. In this paper we are going to discuss about TF-IDF, TextRank and RAKE performance in keyword extraction. The performance is evaluated on statistical parameter precision, recall and F-measure.

II. LITERATURE SURVEY

In this section we mainly concentrated on survey in the field of keyword extraction algorithms performance and text summarization application. The literature survey helped us to get previous proposed algorithms and their working procedure. Keyword extraction is the information retrieval from a text document is part of text mining. The following shows how the authors had proposed their view of point on text summarization.

Kumar, Akshi, et al.[1] has compared the three keyword extraction algorithms; they are TextRank, LexRank, Latent Semantic Analysis. The analysis of keyword extraction algorithms were investigated based on time scale, performance from set of articles, they used ROUGH evaluation matrices for comparison. From all three algorithms the TextRank performs well in comparison.

DA Audich, et al.[17] has presented a systematic study of keyword and key phrase extraction in the online privacy policies. They took four popular keyword extraction algorithms with comparison to human summarization; they got result that TF-IDF performs well in comparison with the manual keyword extraction on small dataset, whereas the AlchmyAPI performs marginally better than TF-IDF on large amount of dataset. The other algorithms RAKE, TextRank outperformed in comparison with basic marginally algorithms on large dataset. They used natural language processing for keyword and key phrase extraction algorithms from a large set of documents and it shows that the keyword and key phrase extraction remains as the difficult task which occurs as the major challenge in the text mining. They used traditional evolution matrices precision, recall and F-measure, calculated the Jaccard's similarity co-efficient between the manual keyword extraction and obtained result from set of data to know the similarity between them.

Liu, Zhengyang, et al.[4] has introduced a PageRank keyword extraction algorithm on Synonym networks. They used a single document as a dataset, the content presented in a single document is marked as a weighted synonym co-occurrence network later the PageRank algorithm is applied on this synonym network to give a rank for created synonym group and also is shows how successfully it can work on natural language network. The PageRank formula executes on the network for several iterations until the score convergence, but it becomes difficult to choose an exact iteration times so the iteration time with different values is compared by keeping default window size 10, as comparison shows that the PageRank algorithm achieves convergence after 20-39 iteration. Similarly the better window size is calculated; by comparison of values the window size of 10 is remains default by giving better output values. The standard evolution matrices precision, recall and F-measure are used and compared between KEA and proposed algorithm, by comparison the proposed algorithms performs batter than KEA on the corpus available in blog pages.

Li, Wengen, et al.[13] in this paper the author proposed TextRank algorithm which is graph based algorithm, the author took Wikipedia as corpus dataset. The short keywords are extracted from the Wikipedia as it consists of large data, the procedure of proposed algorithm is to enrich the short text. The similarity of words is measured by the concept vector and thereafter the construction of keyword the TextRank algorithm is used for keyword extraction. By considering the precision, recall and F-measure value the proposed method of TextRank exploiting on Wikipedia shows better result in comparison with the traditional TextRank, and baseline algorithm TF-IDF for keyword extraction from dataset.

III. PROPOSED ALGORITHMS

This section consists of keyword extraction algorithms, these algorithms are compared to show their effectiveness in the keyword extraction from given corpus. We used following keyword extraction algorithms.

A. TextRank

TextRank keyword extraction algorithm is a popular graph based algorithm and it is the unsupervised algorithm. The TextRank algorithm is the application of the PageRank. The most advanced algorithm for keyword and sentence extraction. The graph is built using natural language processing which gives the relationship between the entities of the text. Vertex gets its weightage if it forms more number of links with other vertices so that it casts a vote in support of that vertex. Thus the score of the vertex is calculated by considering the inbounds and outbounds of the vertex. [1]. The graph is based on ranking of the vertices in the graph can be determined by evaluating the associated score of each vertex. The score of keyword shows the strength and importance of keyword in text document[13]. The score(R) of the vertex V_i is defined by the following equation:

$$R(Vi) = (1-d) + d \cdot \sum j \in I(Vi) \frac{1}{|O(Vj)|} R(Vj) \quad (1)$$

Where G=(V,E) represents a directed graph with the set of vertices V and set of edges E. $I(V_i)$ and $O(V_i)$ represents the in-bounds and out-bounds for a vertex V_i and d ϵ [0,1] is a damping factor in which d is a probability of jumping from one vertex to some random vertex. The damping value generally considered as 0.85.

B. TFIDF

TF-IDF is the baseline, common and most popular keyword extraction algorithm available in text mining. TF is the term frequency, the frequency of each word in text document and the number of words easily determined in text document. IDF is inverse document frequency of word in text document, where the idf is the computation from log of inverse probability of word obtained easily in text document.[20][21].

The term to term weighting and term to sentence weighting can be determined by TF-IDF(term frequency and inverse document frequency). Let *tfi* be the frequency of the term *i* in the entire collection, let *dfi* be the frequency of documents in which *i* occurs, and let *d* be the number of documents in the whole collection. The following equations define the *idf* and *tf-idf* weighting schemes:

$$idf = \log[\frac{d}{dfi}] \quad (2)$$

$$W_{ij}: tf - idf = \text{tfi} \times \text{idfi} \quad (3)$$

C. RAKE

RAKE is the Rapid Automatic Keyword Extraction, the RAKE algorithms is unsupervised method and its independent of domain, language, size of data set method for keyword extraction from single document[19]. In this RAKE application a list of stopword, list of keyphrases or set of words delimiters is taken as the input and later it uses these list to partition the document into candidate keywords. The document text is split into set words by delimiters; the

obtained set of words is then split into series of adjoining words at phrase delimiters and stopword positions. Candidate words are considered as a word within a series which are assigned the similar position in text document and are presented together. The sliding window is not required as like in the TextRank, co-occurrence of words available within the obtained candidate keywords are meaningful and permits to find word co-occurrence without depending on the window size[17][19]. As similar to other keyword extraction algorithm the score for every single candidate keyword is assigned, it can be calculated as the sum of scores of every single of its member term co-occurrences. Keyword scores W(s) are based on its words frequency F(w) and its degree D(w),

$$W(s) = \frac{D(w)}{F(w)} \quad (4)$$

IV. EXPERIMENT

A. *Experimental dataset:*

For proposed keyword extraction algorithms the dataset, we used a single document from large data sets. The document consist of average of 1000 words in total, the same text document is taken as the input to all keyword extraction algorithms.

B. *Pre-processing:*

Pre-processing is initial step in keyword extraction application, we performed various series of steps for normalization of the input text document. The input is primitively converted to lowercase, non-printable words, special character are removed. The python an interpreter language is used, the python consist of huge amount of library package. Python library such as TextBlob, NLTK, Numpy are used to obtain pre-processed dataset.

C. *Algorithm Setup:*

All the algorithms are executed using python. The Textblob is a library used for textual procedure; it is installed for keyword extraction using TF-IDF algorithms. It extracts keywords with the frequency. The TextRank algorithm uses numpy, pandas and NLTK python libraries, in this algorithm implementation many function are defined the public function is exported from the pagerank module and the TextRank module consist of mainly two functions which consist of variable such as document to be passed, damping factor, random surfer probability with constant window size. The window size can be changed as the window size varies it outperforms with different frequency values till certain windowsize.

WindowSize: The width of the window in which two words must fall to be considered to have co-occurred. A word will be considered to have co-occurred with any word one or two words away from it in the document[18].

RAKE algorithm has a python library is used for implementation setup.

D. *Evaluation matrices:*

There do exist numerous evaluation metrics for evaluation of performance of the keyword extraction algorithms due to the difference in keywords weightage and the number of keywords in each document. Usually the traditional, standard statistical evaluation matrices are used those are precision, recall and F-measure[10]. These are well known matrices used for keyword extraction, information retrieval, classification and many more areas in text mining.

Precision: Precision (PRE) is evaluated by finding the difference between accepted value and the obtained value by experiment, then dividing by the obtained value. As the accepted value is the average of the total manually selected keywords and obtained value is machine selected value from the given single text document.

$$PRE = \frac{|(accepted\ value) \cap (obtained\ value)|}{|(obtained\ value)|} \quad (5)$$

Recall: Recall(REC) is the difference between the difference between accepted value and obtained value later dividing by the accepted value.

$$REC = \frac{|(accepted\ value) \cap (obtained\ value)|}{|accepted\ value|} \quad (6)$$

F-measure: The f-measure or f-score is the mean of precision and recall . The f-measure value is calculated using following Equation

$$F - measure = \frac{2 \times PRE \times REC}{PRE + REC} \quad (7)$$

E. *Results and comparison:*

Table 1. Experimental Results

	TF-IDF	RAKE	TextRank
Precision	**20.40%**	**11.10%**	**15.13%**
Recall	**25.21%**	**60.18%**	**13.07%**
F-measure	**22.57%**	**19.97%**	**15.59%**

The Table 1 shows the compared vales. The keywords generated by algorithms were compared with each by using the statistical evaluation framework, as TF-IDF performs gives the precision of 20.40% whereas the RAKE and gives the 11.10% and 15.13%. And the Recall calculation gives RAKE performs better than other two algorithms. As in the F-measure evaluation the TF-IDF outperforms than RAKE and TextRank.

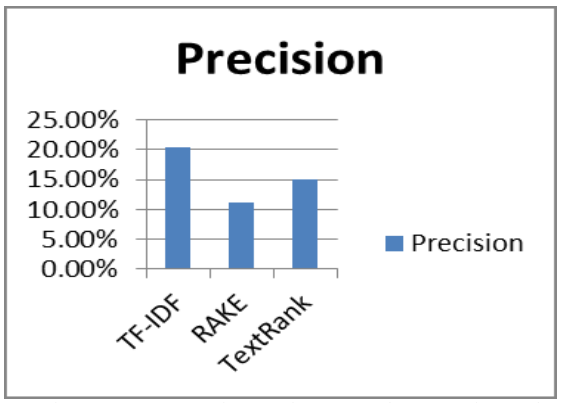

Figure 1. Comparison of Keyword Extraction Algorithms evaluation based on precision

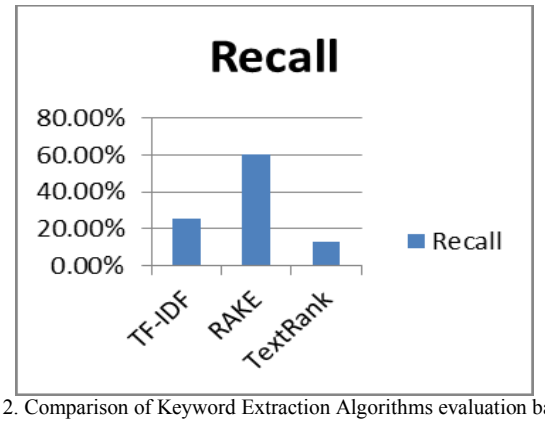

Figure 2. Comparison of Keyword Extraction Algorithms evaluation based on Recall

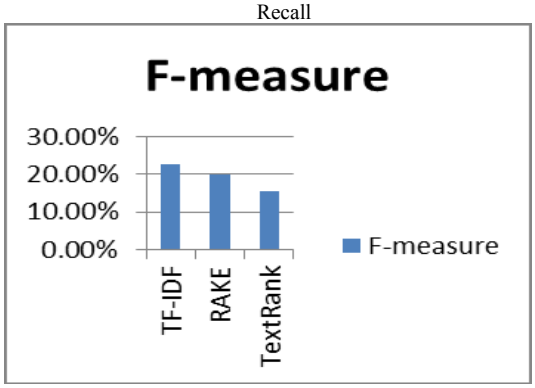

Figure 3. Comparison of Keyword Extraction Algorithms evaluation based on F-measure

The Figure.1 shows the variation in the algorithms. The RAKE generates the important and weighted keywords than TF-IDF. The TextRank extracts keywords with highest score. The Figure2 shows the recall graph, the RAKE achieves with the value 60.18%. The Figure 3 shows the F-measure graph the TF-IDF algorithm achieves the best F-measure in small dataset i.e, a single text document.

V.CONCLUSION

In this paper we have introduced a formal study on keyword extraction algorithms for a single document. The three keyword extraction algorithms were implemented and compared and evaluated on precision, recall and f-measure as the fig 1, 2and 3 presents. The RAKE outperforms over the other algorithms, as the TF-IDF is base line algorithm generates large quantity of keywords but without quality of keywords. The TextRank and RAKE generates weighted and important keywords from a text document. Further we are testing on multiple documents, to test the efficiency of these algorithms.

REFERENCES

[1] Kumar, Akshi, et al. "Performance analysis of keyword extraction algorithms assessing extractive text summarization." *Computer, Communications and Electronics (Comptelix), 2017 International Conference on*. IEEE, 2017.

[2] Zaman, A. N. K., Pascal Matsakis, and Charles Brown. "Evaluation of stop word lists in text retrieval using Latent Semantic Indexing." *Digital Information Management (ICDIM), 2011 Sixth International Conference on*. IEEE, 2011.

[3] Lenjewer, B. "Automatic text summarization with context based keyword extraction." *International Journal of Advance Research in Computer Science and Management Studies* 3.9 (2015).

[4] Yadav, Nidhika, and Niladri Chatterjee. "Text Summarization Using Sentiment Analysis for DUC Data." *Information Technology (ICIT), 2016 International Conference on*. IEEE, 2016.

[5] Dave, Harsha, and Shree Jaswal. "Multiple Text Document Summarization System using hybrid Summarization technique." *Next Generation Computing Technologies (NGCT), 2015 1st International Conference on*. IEEE, 2015.

[6] Moratanch, N., and S. Chitrakala. "A survey on extractive text summarization." *Computer, Communication and Signal Processing (ICCCSP), 2017 International Conference on*. IEEE, 2017.

[7] Lenjewer, B. "Automatic text summarization with context based keyword extraction." *International Journal of Advance Research in Computer Science and Management Studies* 3.9 (2015).

[8] Wu, Kang, Ping Shi, and Da Pan. "An approach to automatic summarization for chinese text based on the combination of spectral clustering and LexRank." *Fuzzy Systems and Knowledge Discovery (FSKD), 2015 12th International Conference on*. IEEE, 2015.

[9] Moratanch, N., and S. Chitrakala. "A survey on abstractive text summarization." *Circuit, Power and Computing Technologies (ICCPCT), 2016 International Conference on*. IEEE, 2016.

[10] Ghosh, Partha, and Soumya Sen. "Time and location based summarized PageRank calculation of Web pages." *Industrial Technology (ICIT), 2014 IEEE International Conference on*. IEEE, 2014.

[11] Tu, Hong T., Tuoi T. Phan, and Khu P. Nguyen. "An adaptive Latent Semantic Analysis for text mining." *System Science and Engineering (ICSSE), 2017 International Conference on*. IEEE, 2017.

[12] Mihalcea, Rada, and Paul Tarau. "Textrank: Bringing order into text." Proceedings of the 2004 conference on empirical methods in natural language processing. 2004.

[13] Li, Wengen, and Jiabao Zhao. "TextRank algorithm by exploiting Wikipedia for short text keywords extraction." *Information Science and Control Engineering (ICISCE), 2016 3rd International Conference on*. IEEE, 2016.

[14] Page, Lawrence, et al. The PageRank citation ranking: Bringing order to the web. Stanford InfoLab, 1999.

[15] Chang, Te-Min, and Wen-Feng Hsiao. "A hybrid approach to automatic text summarization." *Computer and Information Technology, 2008. CIT 2008. 8th IEEE International Conference on*. IEEE, 2008.

[16] Qin, Ying. "Applying frequency and location information to keyword extraction in single document." *Cloud Computing and Intelligent Systems (CCIS), 2012 IEEE 2nd International Conference on*. Vol. 3. IEEE, 2012.

[17] Audich, Dhiren A., Rozita Dara, and Blair Nonnecke. "Extracting keyword and keyphrase from online privacy policies." *Digital Information Management (ICDIM), 2016 Eleventh International Conference on*. IEEE, 2016.

[18] Liu, Zhengyang, et al. "Keyword extraction using PageRank on synonym networks." *E-Product E-Service and E-Entertainment (ICEEE), 2010 International Conference on*. IEEE, 2010.

[19] Rose, Stuart, et al. "Automatic keyword extraction from individual documents." *Text Mining: Applications and Theory*(2010): 1-20.

[20] Dadgar, Seyyed Mohammad Hossein, Mohammad Shirzad Araghi, and Morteza Mastery Farahani. "A novel text mining approach based on TF-IDF and Support Vector Machine for news classification." *Engineering and Technology (ICETECH), 2016 IEEE International Conference on*. IEEE, 2016.

[21] Hakim, Ari Aulia, et al. "Automated document classification for news article in Bahasa Indonesia based on term frequency inverse document frequency (TF-IDF) approach." *Information Technology and Electrical Engineering (ICITEE), 2014 6th International Conference on*. IEEE, 2014.