An Overview of Techniques Used for Extracting Keywords from Documents

Menaka S, Radha N

Research Scholar, Assistant Professor
Department of Computer Applications
PSGR Krishnammal College for Women, GRG School of Applied Computer Technology
Peelamedu, Coimbatore

***Abstract*— Keywords are a set of major words in a document that give high-level description of the content for readers. Keywords are useful for scanning large documents in a short time. Extracting keywords manually are very difficult and time-consuming process. Therefore, there is in need for process to extract keywords from documents automatically. Keyword extraction is a process in which a set of words are selected that gives the meaning of the whole document. This paper presents an overview of techniques used for keyword extraction.**

Keywords*— TF-*IDF, Classification, Lexical chain, WordNet.

I.Introduction

Keyword is the smallest unit, which expresses meaning of entire document that also used for extracting exact information as per user requirements. Everyday thousands of books, papers, articles and documents are created and published. It is very difficult to go through all the text materials, so that there is a need of good information extraction or summarization method that provides the real contents of a given document. Various applications [1] can take advantage of it such as information retrieval, automatic indexing, text summarization, classification, clustering, topic detection and tracking, web searches, report generation, filtering, cataloging, etc.

Keyword extraction, also known as keyphrase extraction is an area of text mining that intends to identify the most useful and important words, phrases that are also called terms. Keyword extraction is an essential technique for web page retrieval, document retrieval, document clustering, text mining, and so on. The basic idea is to select words from a text that gives a good thought to its content. Keyword extraction from the documents includes so many processes. First process is to select the documents; the documents can be of text or html, or pdf, etc. The next process is to pre-process the document that involves removing the stop words, stem the words. After pre-processing the keywords are extracted by using the extraction techniques.

There are various keyword extraction approaches such as statistics approach, linguistic approach, machine learning approach, etc are used to extract keywords from the document. Machine Learning techniques consider the keyword extraction as a classification problem. WordNet dictionary also used to find the similarity between words to extract most important keywords. In the following sections various techniques for selecting effective keywords are elaborated.

II. Literature Survey

Keyword extraction is a process used to mine limited number of words from documents. This extraction process should be done in a systematic way and by at least or no human interferences. Statistical methods are simple and don't need training data. The methods such as term frequency, word co-occurrence, TF-IDF, N-gram are statistical methods. Salton et.al [2] used N-Gram method for automatic document indexing.

Linguistic methods use words, sentences and documents linguistic features such as part of speech, syntax and semantic. Hulth et.al [3] examined different methods in keyword extraction using linguistic features. Term frequency, inverse document frequency and relative position of use of keywords and part of speech label are used. This will significantly improve automatic keyword extraction process.

Rose et.al [4] used Rapid Automatic Keyword Extraction. RAKE is an unsupervised, language-independent and domain-independent method for extracting keywords from individual documents. Rogina et.al [5], extract keywords from lecture slide, and then used as queries to retrieve relevant web documents.

Frank et.al [6] used machine learning techniques to improve keyword extraction process. Various methods of machine learning are available which are more complex and have higher computational cost. Suzuki et.al [7] applied natural language processing techniques for keyword extraction from radio news. Wikipedia, Encyclopedia and journal papers were used as resources to determine the keywords relations.

III. Keyword Extraction Using Lexical Chain

A lexical chain is a sequence of related words in writing, spanning short (adjacent words or sentences) or long distances (i.e., entire text). A chain is independent of the grammatical structure of the text and in effect it is a list of words that captures a portion of the cohesive structure of the text.

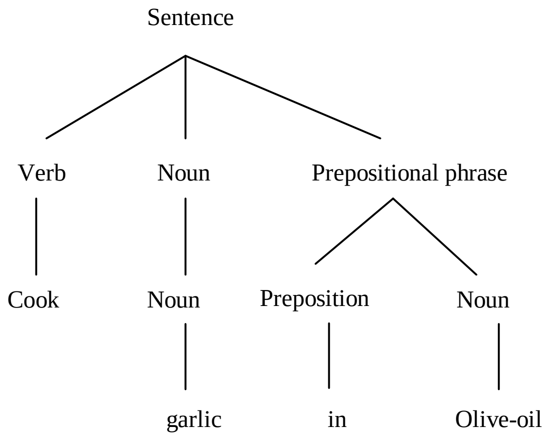

Fig 1 Example for lexical chain

Fig. 1 shows an example of lexical chain. Lexical chains are used in different NLP problems such as word sense disambiguation, text segmentation, text summarization and topic tracing. WordNet dictionary is used to build the lexical chain that provides the word senses and semantic relations between words. Lexical chain builder uses WordNet relations, which are Synonym, Hypernym or Hyponym, Meronym to build a lexical chain.

1) Synonym: x denotes the same as y, i.e., the semantic relation that holds between two words.

2) Hyponym: x is subordinate of y or “has-property”, subordination - the semantic relation of being subordinate or belongs to a lower rank or class.

3)Meronym: x is a part of y, i.e., the semantic relation that holds between a part and the whole.

Every node in a lexical chain denotes a meaning of a word, and each link can be synonym, hyponym or hypernym, or meronym relation between two word senses. In this approach keywords are extracted using the following features that are,

a) First occurrence position
b) Word frequency
c) Last occurrence position
d) Lexical chain score of a word
e) Direct lexical chain score of a word
f) Lexical span score of a word
g) Direct lexical span score of a word

Lexical Chain Score of a Word

A word can be a member of more than one lexical chain. The score can be assigned for these words. Then the word that has the maximum score is chose as the lexical chain score of the word. The score depends on the relations appearing in the lexical chain.

Direct Lexical Chain Score of a Word

This can be calculated by scoring only the relations that belong to the word.

Lexical Span Score of a Word

The span score of a lexical chain depends on the portion of the text that is covered by the lexical chain. This covered portion of the text is considered to be the distance between the first occurrence position of a lexical chain member (word) and the last occurrence position of a lexical chain member (word). The span score is computed by finding the difference between these two positions.

Direct Lexical Span Score of a Word

The score of the lexical chain with maximum score can be considered as the direct lexical chain span score of the word. This score can be computed as same as the lexical chain span score except that the words that are directly related with the word in the lexical chain.

This technique uses a statistical classifier (C4.5) to build decision trees that is to identify whether the word is likely one or not. The decision tree uses bagging; it is a process of classifying the objects with multiple classifiers. In bagging technique the average classification probability is used to classify the objects. Gonenc Ercan, Ilyas Cicekli has proposed this approach [8] with a corpus for extracting keywords. Precision values are calculated for this system with all seven features gave the better results. Wikipedia also used to relate phrases in the lexical chain.

IV. Word Co-occurrence

Word co-occurrence is extensively used in various forms of research such as content analysis, text mining, construction of thesauri, ontology’s, etc. Its aim is to find similarity between words or similarities of meaning among word patterns. The sentences in the document are considered as a set of words; it includes title of a document, section title and a caption.

The term frequency is determined by counting the frequent terms occurred in a document. The frequencies of the co-occurred term can be represented in N×N matrix format. Co-occurrence distribution [9] shows the importance of terms in a document and the co-occurrence biases are derived from semantic or lexical or from other relations.

Clustering methods are also used for this approach to cluster the frequent terms. The terms are clustered using similarity distribution of co-occurrence with other terms. Co-occurrence terms are counted from these clusters and then the expected probability is calculated. Then the statistical value of X^2 is used to measure the degree of biases of distribution, which is calculated using the formula,

$$X^2 = \sum_{g \in G} \frac{(freq(w,g) - n_w\, p_g\,)^2}{n_w\, p_g} \qquad ------ (1)$$

In this freq(w,g) denotes frequency of co-occurrence of term w and g. (freq(w,g)-$n_w p_g$) denotes the difference between predictable frequencies. $n_w p_g$ represents the expected frequency of co-occurrence, in which n_w represents the total number of terms in the sentence where w appears and p_g denotes the sum of the total number of terms where g appears is proportional to the total number of terms in the document.

To measure the robustness of the X^2 value, by subtracting the maximal term with it. If the X^2 value is high then it is considered as the important word in the document.

Yutaka Matsuo, Mitsuru Ishizuka [10] proposed this approach with 20 technical papers. Top 15 words are selected from each paper by using TF, TF-IDF, Keygraph and word co-occurrence methods. Then the authors check that the terms are important to the documents or not. The precision can be calculated by ratio of the checked terms to the selected 15 terms. In this coverage of each method is calculated the indispensable terms included in the 15 terms to all the indispensable terms.

V.Graph Based Keyword Extraction

A graph has been built after doing the basic text pre-processing operations such as stemming and stopwords removal. Only a single vertex for each distinct word is created even if it appears more than once in the text. Thus each vertex label in the graph is unique. There is a directed edge from the vertex corresponding to the term *x* to the vertex corresponding to term *y*, if a word *x* immediately precedes a word *y* in the same sentence somewhere in the document.

An edge cannot be created when the sentence terminating punctuation marks are present between two words. Each distinct word in a text is represented as a node in the document graph. In this, both supervised and unsupervised approaches are used.

The nodes of document graphs are identified by training a supervised approach such as classification algorithm on a repository of summarized documents. Each node of every document graph belongs to one of two classes are,

1) YES if the corresponding word is included in the document extractive summary and
2) NO otherwise.

The features used for extracting keywords from a document are as follows:

- In Degree - number of incoming edges
- Out Degree - number of outgoing edges
- Degree - total number of edges
- Frequency - term frequency of word represented by node
- Frequent words distribution ∈{0,1}, equals 1 iff Frequency >= threshold[2]
- Location score- calculates an average of location scores between all sentences containing the word N represented by node

$$Score(N) = \frac{\sum_{S_i \in S(N)} Score(S_i)}{|S(N)|} \quad ----(2)$$

- Tf-Idf Score – calculates the tf-idf score of the word represented by node.
- Headline Score ∈ {0,1}, equals to1 iff the document headlines contains word represented by node.

The unsupervised approach such as ranking algorithm is used to extract the keywords from the documents. The authors Marina Litvak and Mark Last [11] have performed this process on the collection of summarized news articles provided by the DUC (Document Understanding Conference).

In supervised approach authors used several classification algorithms such as C4.5, Support Vector Machine and Naive Bayes are implemented in Weka software for building classification models algorithm to identify whether a word belongs to document or not. They get better results using Naive Bayes classification algorithm.

In unsupervised approach, HITS ranking algorithm is used to document graphs and evaluate its performance on unsupervised text extraction. The HITS algorithm distinguishes between authorities (i.e., pages with a large number of incoming links) and hubs (i.e., pages with a large number of outgoing links). HITS algorithm produces an authority score and hub score for each node. For the total rank (H) calculation they used the following four functions:

1) H (Vi) = HITSA (Vi)
2) H (Vi) = HITSH (Vi)
3) H (Vi) = avg {HITSA (Vi), HITSH (Vi)}
4) H (Vi) = max {HITSA (Vi) , HITSH (Vi)}

where, HITSA(Vi) denotes the authority score and HITSH(Vi) denotes the hub score. Authors have compared the results of both supervised and unsupervised approach and conclude that the supervised approach is the most accurate option for identifying keywords in a document graph.

VI. NEURAL BASED APPROACH

Neural network model is used to find the keywords from the document. Features will be defined and the architecture of the back propagation is designed for judging keywords. Before determining feature values, a group of documents is selected and used to find the two features for each word are Inverted Document Frequency [IDF] and Inverted Term Frequency [ITF]. The input features required for this approach are:

1) TF (Term Frequency): TF is the frequent occurrence of the word in single document.
2) IDF (Inverted Document Frequency): IDF is the measure of importance of the word in the sample documents.
3) ITF (Inverted Term Frequency): ITF denotes the total frequency of the word in sample documents.
4) T (Title): T denotes the existence of the word in the title of the given document.
5) FS (First Sentence): FS denotes the existence of the word in the first sentence of the given document.
6) LS (Last Sentence): LS denotes the existence of the word in the last sentence in the given document.

The features, TF, IDF, and ITF, are represented in integers as greater than or equal to zero, while the others, T, FS, and LS, are represented in binary values as zero or one. For example, if the word is in the title of the document, T is one otherwise T is zero. The output features for each word is represented in binary values which are,

1) K (Keyword): If the word is judged as keyword, K is one, otherwise zero

2) N (Non-keyword): If the word is judged as non-keyword, K is zero, otherwise one.

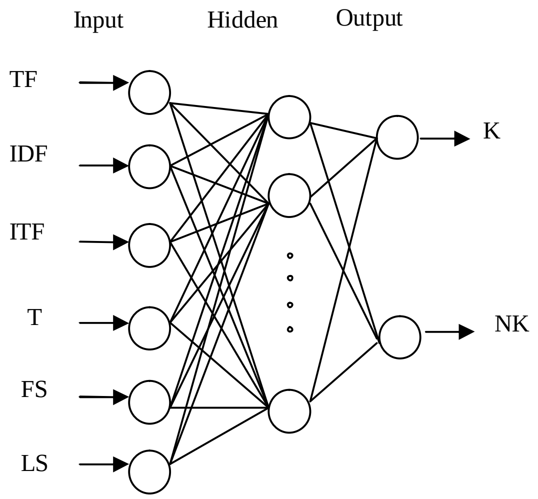

Fig 2 The Architecture of Back Propagation to Judge Keywords

Fig 2 shows the architecture of back propagation for extracting keywords. Author Taeho Jo [12] proposed this technique to extract keywords from news articles. The neural based approach to judge keywords is compared with equations based on TF (Term Frequency) and IDF (Inverse Document Frequency).

$$W_i = TF(\log_2 N - \log_2 IDF + 1) \qquad ------(3)$$

$$W_i = \frac{TF^m}{(IDF + ITF + 1)^n} \qquad ------(4)$$

The equation (3) is used to calculate the weight of each term in the document. Then the equation (4) is used to develop two modules are text categorization and text summarization. ITF in the equation is the total frequency of words in the sample documents and TF is the total frequency of the word in a specified document. The precision for judging keywords in documents with this approach is maximized, when the threshold value is given as maximum and is trained with maximum epochs.

TABLE I

TECHNIQUES FOR KEYWORD EXTRACTION

Author	Technique	Dataset	Precision (in %)
Gonenc Ercan, Ilyas Cicekli	Lexical chain	75 Journal articles (abstracts)	45
Yutaka Matsuo, Mitsuru Ishizuka	Word co-occurrence	20 Technical papers	51
Marina Litvak, Mark Last	Graph-based approach	Summarized news articles	Above 50 percent
Taeho Jo	Neural-based approach	900 News articles	92

VII. CONCLUSION

This paper represents various techniques available for extracting keywords from the documents. Keywords are used to define, revise, remember, share and choose the learning objects to read easily. The approaches described above shows different ways to extract the efficient keywords from documents. TF-IDF method is used in most of the approaches to identify the frequency of the words. Neural based approach provides a better precision value when compared to other approaches stated above. In future neural-based approach or some other approaches can be used for extracting the keywords from documents.

REFERENCES

[1] D.B. Bracewell, F. REN, S. Kuriow. 2005 "*Multilingual Single Document Keyword Extraction for Information* Retrieval", in Proceedings of Natural Language Processing and Knowledge Engineering, p. 517-522.

[2] Salton G. 1989. "*Automatic text processing*", published in Addison-Wesley Longman publications.

[3] Hulth A. 2003 "*Improved automatic keyword extraction given more linguistic knowledge*", in Proceedings of the conference on Empirical methods in natural language processing, p.216-223.

[4] Stuart Rose, Dave Engel, Nick Cramer and Wendy Cowley. 2010 "*Automatic Keyword Extraction from Individual Documents*", in Text mining: Applications and Theory, p. 3-20.

[5] Rogina I, Schaaf T.2002 "*Lecture and presentation tracking in an intelligent meeting room*", in Proceedings of 4th International Conference on Multimodal Interfaces, p. 47-52.

[6] Frank E., Paynter G.W., Witten I.H., Gutwin C., & Nevill-Manning C.G. 1999 "*Domain-specific keyphrase extraction*", in Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence.

[7] Suzuki Y, Fukumoto F, Sekiguchi Y. 1998 "*Keyword extraction of radio news using term weighting with an encyclopedia and newspaper articles*", SIGIR, 1998.

[8] Gonenc Ercan, Ilyas Cicekli. 2007 "*Using Lexical Chains for Keyword Extraction*", published in Information Processing and Management, Volume 43 Issue 6, p. 1705-1714.

[9] Christian Wartena, Brusee, Slakhorst. 2010 "*Keyword Extraction using Word Co-occurrence*", published in Database and Expert System Applications, p. 54 - 58.

[10] Yutaka Matsuo, Mitsuru Ishizuka. 2003 "*Keyword Extraction from a Single Document using Word Co-occurrence Statistical Information*", published in International Journal on Artificial Intelligence Tools, Volume13, Issue1, p. 157-169.

[11] Marina Litvak, Mark Last. 2008 "*Graph-Based Keyword Extraction for Single-Document Summarization*", published in Workshop on Multi-source Multilingual Information Extraction and Summarization, p.17-24.

[12] Taeho Jo. 2003, **"***Neural Based Approach to Keyword Extraction from Documents*". Proceedings of International conference on Computational science and its applications in Canada, part 1, p. 456-461.

[13] Mihalcea R. 2004 "*Graph-based ranking algorithms for sentence extraction, applied to text summarization*", in Proceedings of the 42nd Annual Meeting of the Association for Computational Lingusitics.

[14] Bollegala D, Matsuo Y, Ishizuka M. 2006 "*Extracting key phrases to disambiguate personal names on the web*", in Proceedings of 7th International conference on Computational Linguistics and Intelligent Text Processing, p. 223-234.

[15] Song Y, Huang J, Councill I.G, Li J, Giles C.L. 2007 "*Efficient topic-based unsupervised name disambiguation*", in Proceedings of Joint Conference on Digital Libraries, p. 17-23,