A Statistical Approach of Keyword Extraction for Efficient Retrieval

Shruti Luthra
B.Tech (Student)
BPIT-Delhi

Dinkar Arora
B.Tech (Student)
BPIT-Delhi

Kanika Mittal
Assistant Professor
BPIT-Delhi

Anusha Chhabra
Assistant Professor
BPIT-Delhi

ABSTRACT
Large number of techniques for keyword extraction have been proposed for better matching of documents with the user's query but most of them deal with tf-idf to find the weight age of query terms in the entire document but this can result in improper result as if a term has a low term frequency in overall document but high frequency in a certain part of the document then that term can be ignored by traditional tf-idf method. Through this paper, the keyword extraction is improved using a hybrid technique in which the entire document is split into multiple domains using a master keyword and the frequency of all unique words is found in every domain . The words having high frequency are selected as candidate keywords and the final selection is made on the basis of a graph which is constructed between the keywords using Word Net. The experiments, conducted on various documents show that proposed approach outperforms other keyword extraction methodologies by enhancing document retrieval.

Keywords
Information Retrieval, Domain Splitting, Natural Language Processing, Inverse Document Frequency, Word Net

1. INTRODUCTION
Information retrieval [1] in a process of extracting information resources which are required for the retrieval of useful information from multiple sources. Some automated information retrieval systems have been introduced in order to reduce the overloading of information. There are many colleges and libraries which use IR systems to provide access to multiple books, papers, articles, journals and other documents. In information retrieval, there is a collection of documents from which index terms are extracted from each and every single document. Sometimes information retrieval systems are not able to give efficient result as the polysemous meanings, or senses, of words can lead to keyword queries that are ambiguous. IR systems are unable to differentiate between the different senses of a keyword and retrieve documents that contain all the senses. Thus the problem is assisting the user in clarifying and analyzing the problem and determining information needs [2].

Keywords play a very important role in the extraction of relevant information. With the help of a few keywords, the meaning or the summary of a document can be extracted. Since keyword express meaning of entire document and is the smallest unit , many applications can take advantage of it such as text summarization automatic indexing , information retrieval , classification clustering ,filtering ,topic detection and tracking, cataloging , information visualization ,web searches, report generation,getting the context of document etc.[3].

Some of the methods for Automatic Keyword Extraction are statistics, linguistic, machine learning[3].In **Statistics Approach** the statistical information of the words are used to search for the keywords in a document while in **Linguistic Approach** the nouns are considered as they contain a large amount of information. The nouns phrases are extracted and scored after its morphological analysis. The **Machine Learning Approach** trains the machine in such a way that it can find keywords in a document. It simply employs the model from its previous experience to extract keywords. Some **Other Approaches** for keyword extraction combines the methods given above and use some previous knowledge such as the position of the words, length of the words, layout feature of words, html tags around of the words, etc. But these techniques have various drawbacks as position of words fails to consider a keyword important if it is in appears in middle of a paragraph which may represent the document rather it will chose a keyword from the heading section. The length of word method would not consider any word below its word length limit so if word length limit is quite high many short words would not even be considered.

Another method for extraction of keywords is **Term Frequency-Inverse Document Frequency** [4]. The purpose of TF-IDF weight is to evaluate the importance of a word in a document from a collection of words. The importance of a word is directly proportional to the number of times it occurs in a document. Inverse Document Frequency [5] is a measure to evaluate the importance of a term. It is done by dividing the total number of documents by the number of documents containing the term and the taking logarithm of the quotient. TF-IDF faces drawbacks like if a word occurs multiple times in a single paragraph but not in the overall document TF-IDF will not consider the word as keyword considering its low frequency with respect to overall document. Such words can represent an important context of document. Keywords can also be extracted by using graph based approaches [6]. Graph can be made by taking words as vertices and edges as relationships among words. Edges can be established among words on various principles.

Through this paper we will propose a method to overcome the limitations of the conventional TF-IDF formula that are mentioned above by using the concept of domain splitting. The algorithm proposed in this paper is a combination of both semantic and syntactic analysis. In section II of the paper the some of the work that has been done till now in the field of keyword extraction is discussed. In section III, we will explain the proposed methodology along with a flowchart. Finally in section IV we will conclude our research.

2. RELATED WORK
Various algorithms have been proposed by the researchers for keyword extraction which can be classified into three classes [7]: simple statistics, linguistics and machine learning-based. Simple statistics based approaches, have limited prerequisites, simple to understand than other approaches and they focus on non-linguistic features of the text. The keywords in the

document can be identified using the statistical information of the words. Cohen [8] used N gram statistical information to index the document. Other statistical methods that are used for keyword extraction include word frequency, term frequency [9], word co-occurrences etc. The statistical methods generate useful results and are easy to understand.

Jasmeen et. Al in [10] talked about various statistical approaches for keyword extraction like identify noun phase as a keyword where the nouns are identified, scores are given to them and they are then recognized as keywords. Another method proposed was position weight where a word is given score on the basis of where it is used in the document and the words with high scores are considered as keywords.

Other class is of Linguistics approaches. They use the linguistic features of the words, sentences and document. They are more complicated than the statistical approaches as linguistic features of the words are given more importance. Hulth [11,12] examined various techniques so as to incorporate keyword extraction and linguistics features of words. It is seen from the experimental results use of linguistic features significantly improve the performance of the automatic keyword extraction.

The other class is machine learning algorithm. The machine learning based algorithms work as: First a set of training documents is selected and is then given to the system, a range of keywords is given for each document. Then this knowledge that is gained from set of documents is applied to the documents to extract keywords from them. These methods used naive Bayes formula, support vector machine [13] for domain-based extraction of technical key phrases. Suzuki et al. [14] used spoken language processing techniques to extract keywords from radio news.

The approaches for keyword extraction can be also be categorized into either (1) **unsupervised** or (2) **supervised**. Supervised approaches require annotated data source to process, while unsupervised require no such annotations in advance.

The main idea that is followed for supervised methods is to transform keywords extraction into a binary classification task: Kea (Witten et al., 1999 [15]) and GenEx (Turney, 1999 [16]) are two typical and well-known systems [15, 16], which set the whole research field of the keyword extraction. The task is to classify words that are given into keywords candidates, which is a binary classification task word is either keyword or not.

In [11] as stated earlier Hulth uses Noun Phrase chunks (NP) (rather than term frequency and n-grams),and explores incorporation of the linguistic knowledge into the extraction of keywords and, and as a feature adds the POS tag(s. In more details,by adding POS tag(s) given to them improves the results independent of the term selection approach applied and extracting NP-chunks gives better results than n grams.

Nguyen and Kan (2007) [17] propose algorithm for keyword extraction from scientific publications using linguistic knowledge. They introduced features that can capture salient morphological phenomena found in scientific key phrases, such as whether a candidate key phase that is selected is an acronym or if it uses specific terminologically productive suffixes.

NLP techniques were used by Krapivin et al. (2010) in [18] to consider machine learning approach and improve them (SVM, Local SVM, Random Forests) to solve the problem of automatic keyphrases extraction from scientific papers. Evaluation showed efficient results that can that outperform state-of-the-art Bayesian learning system KEA on the same dataset without the use of controlled vocabularies. HaCohen-Kerner (2003) in [19] presents a simple model that uses uses unigrams, 2-grams and 3-grams, and stop-words list and extracts keywords from abstracts and titles.The model gives the weighting of words and the highest weighted group of words (merged and sorted n-grams) are proposed as keywords.

Supervised and unsupervised were compared by Litvak and Last (2008) in [20] for keywords identification in the process of keyword extraction from document. The basis of the approaches was graph-based syntactic representation of text and web documents. The results of the HITS algorithm on a set of summarized documents performed comparably to supervised methods (Naïve Bayes, J48, SVM). According to the authors simple degree-based rankings from the first iteration of HITS, rather than running it to its convergence, should be considered.

Research by Yang et al. (2013) in [21] focused on entropy difference between the intrinsic and extrinsic modes for keyword extraction, which states that the keyword extracted gives the intention of author about the document. Shannon's entropy difference between the intrinsic and extrinsic mode was used in their method, which refers that words occurrences are modulated by the author's purpose, while the irrelevant words are distributed randomly in the text. This indicates that any natural language document with words clearly identified can apply these ideas, without requiring any previous knowledge about semantics or syntax.

3. PROPOSED WORK

There are various techniques for extracting keywords from a document or a number of documents. The techniques can follow different approaches out of which the most used and efficient approaches are syntactic and semantic approaches[22]. In syntactic approach the words are considered as keywords on the basis of their position in document or number of times it occurs etc. In semantic approach the semantic relationships among words are considered so that they can accurately represent the meaning of documents.

In this paper we propose a technique which is a combination of these two approaches so as to make keyword extraction process more efficient. One of the techniques which use syntactic approach is TF-IDF. The formula for which is as follows[23]

$$tfidf(t,d,D)=tf(t,d)\times idf(t,D) \quad (1)$$

TF-IDF makes assumptions that the keyword that has been selected has a high frequency in the document that is selected i.e., a large TF value and it has a low document frequency in the whole document collection that is a small document frequency value[24] but then if a term has a low term frequency in overall document but high frequency in a certain part of the document the term would not be selected according to TF-IDF but that term could have been an important keyword of the document.

Through our proposed method we have handled the drawbacks of the technique. The method is as follows:

Step 1: Select the training documents D.

Step 2: Remove stop words which are irrelevant words like the, a, with, behind etc appearing frequently.

Step 3: Perform stemming (Porter's Algorithm) [25] on words that is identify words that are syntactic variants of one another and group them.

Step 4: A master keyword (m) is given by the user related to the documents in order to divide the document into separate domains. For example if the documents are on military forces the keyword can be "military".

Step 5: Domains are split on the basis of' 'm' provided by the user and they are split in a way that a window is considered from the place where first time keyword is appearing in the document till the next time it appears.

Step 6: Let the domains be $\{d_1,d_2\ldots\ldots d_n\}$ where n>0 where D is the set of training documents and d_i be the i^{th} domain.

Step 7: Let the number of terms be t_i where i>0.Calculate term frequency (tf) of every term appearing in every domain.

Step 8: Consider two buckets b_1 and b_2.

Step 9: For all the i^{th} terms in every domain calculate threshold frequency (β) which would be the average frequency of all the terms. Select all the terms whose tf>β and put them in bucket b_1.

Step 10: Now consider all terms that appeared in any of the domains and select the terms which appear in min n/2 domains or more and put them in bucket b_2.

Step 11: Combine the terms in b_1 and b_2 removing all the redundant terms in bucket b.

Step 12: Let there be w keywords in bucket b. After performing stemming on documents consider each word in bucket b and take window of w keywords and consider all keywords coming in window and are present in bucket b.

Step 13: Make a graph then by taking all keywords in bucket b as nodes and make connections to every keyword that appeared in it's window in step 12.

Step14: After making the graph calculate α = number of edges/number of vertices and consider all words whose number of edges > α as final keywords.

- Consider Training document d_1-d_n, master keyword "m", term frequency tf, threshold frequency β, queues $q_1,q_2,\ldots.q_n$, array $a_1,a_2\ldots a_N$, array b_1,b_2,
- Start a for loop to find the first occurrence of "m" to the next occurrence of "m".
- Put all the words from first to next occurrence of m in Queue q_1.
- From the second occurrence of "m" find till the third occurrence of "m" and put it in Queue q_2.
- Repeat the process and make domains on the basis of "m" and put it in queue till eof(end of file)
- For every word in q_1 calculate term frequency tf using variable counter.
- Take every item of queue and compare with others and for every occurrence of the word increment counter by one.
- After eof(end of file) put the word and value of counter in a_1.
- Repeat the process for every word in q_1.
- Repeat for every Queue and add values in respective arrays.
- For every array $a_1..a_n$ calculate β which is the average of the values it has.
- For every word tf>=β add words in bucket b_1.
- All words common in more than n/2 domains are added into b_2.
- Remove all common words from b_1 and b_2 and combine all in b.
- Make Graph by the method explained in Step 12 and 13 of proposed method and calculate α.
- Recognize Keywords whose number of edges > α as final keywords.

Fig 1: Proposed algorithm

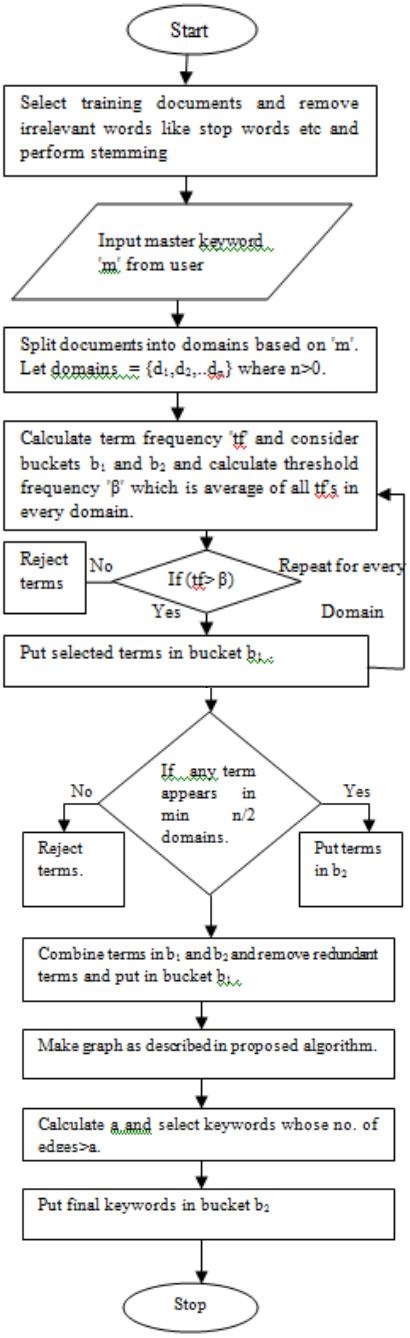

Fig 2: Flow Chart of Proposed Algorithm

My dream car is the Ferrari. It has always been a symbol of speed and prestige and elegance. It is the best machine car in the world. It has even been described as a 'passion on wheels'. Owning it means that you have arrived in style. But of course it is not easy. Only the very rich can even dream of owning it. Its a dream car for every human because of its high speed and quick wheels.It is the brainchild of Enzo Ferrari, an Italian who started his career as a small carmaker and soon took up car racing.The carmaker Enzo designed a completely whole new generation of cars.Enzo Ferrari did not intend to build road cars, when he formed the Scuderia in 1929. He did this as a sponsorship for amateur drivers. After World War I, Ferrari became a driver on the Alfa-Romeo team.Following a victory at Ravenna, the family of Francesco Baracca, a World War I fighter pilot for the Italian air force, presented him with a small charred plaque. Baracca had died in combat and the plaque was one of the surviving remnants of his machine plane. It contained their family crest, a black horse on a yellow shield. This prancing horse became the symbol of Ferrari and it appeared on all the cars he drove. Even today it can be seen on the distinctive red racing and it is known as the best racing car in the world Ferraris and the touring machine cars he makes for the public. The Ferrari has been featured in many films and television shows especially in California. The 250 GT California was seen in "Ferris Bueller's Day Off," the Ferrari 512 was in the 1971 film "Le Mans" with Steve McQueen, the Ferrari "Mondial" was in Weird Science and the Ferrari Daytona appeared in "Miami Vice". In a truly extraordinary and record-setting partnership between man and the extraordinary machine, Michael Schumacher and Ferrari dominated the Formula One races with this extraordinary machine car, winning the World Driver's Championship from 2000 through 2004 and the Constructors' Championship from 1999 through 2004. Among four wheelers, it is simply the ultimate in style and power and that is why it is my dream car.

Fig 3: Example

dream car Ferrari symbol speed prestige elegance passion wheels style course rich dream brainchild Enzo **Ferrari** Italian career small carmaker car racing Enzo **Ferrari** road cars Scuderia sponsorship amateur drivers World War I **Ferrari** driver Alfa-Romeo team
victory Ravenna family Francesco Baracca World War I fighter pilot Italian air force, presented small charred plaque Baracca died combat plaque remnants family crest black horse yellow shield prancing horse symbol **Ferrari** appeared cars drove distinctive red racing Ferraris touring cars public **Ferrari** films television shows 250 GT California "Ferris Bueller's Day Off" Ferrari 512 1971 film "Le Mans" Steve McQueen **Ferrari** "Mondial" Weird Science Ferrari Daytona "Miami Vice" truly extraordinary record-setting partnership man machine, Michael Schumacher **Ferrari** Formula One races World Driver's Championship Constructors' Championship four wheelers simply ultimate style power dream car

Fig 4: Domain Splitting Based on Master Keyword

Illustration of proposed algorithm through Example:.

Consider the following paragraph in figure 3 for extraction of keywords:

Step 1: Master keyword "Ferrari " is given by the user.

Step 2: Domains are made as given in proposed method on the text obtained after stemming of words and stop words removal as shown in figure 4.

Step 3: term frequency is calculated for all words in every domain and words are added to buckets as shown in figures 5 and 6.

Step 4: Similar calculations are done for term frequencies for each domain and results are obtained as in fig 7.

Step 5: Graph is then constructed as given in the proposed method in figure 8.

Domain D1:-

Symbol=1,Speed=2,Prestige=1,Elegance=1Best=1Machine=1 Car=2World=1Passion=1Wheels=2
Style=1Course=1Rich=1Dream=2Human=1High=1Speed=1 Quick=1 Wheels=1 Brainchild=1 Enzo=1
Total number of words=25.Number of distinct words=21Eligible keyword frequency required = Total number of words/ Number of distinct words
= 25/21 = 1.19 =2(absolute)
Hence keywords from domain 1 are speed , cars , wheels and dream.

Fig 5: Calculations for Domain D1

Domain D2:-

Italian=1 Career=1 Small=1 Carmaker=2 Car=2 Racing=1 Enzo=2 Designed=1 New=1 Generation=1
Cars=1
Total number of words=14.,Number of distinct words=11.
Eligible keyword frequency required = Total number of words/ Number of distinct words
14/11 = 1.27 =2(absolute)
Hence keywords from domain 2 are car , carmaker and Enzo.

Fig 6: Calculations for Domain D2

D1={ speed , dream , wheels}

D2= {carmaker , car , Enzo}

D3=None

D4={ Baracca , plaque , horse}

D5= {cars , racing}

D6= {California , film}

D7=None

D8= {extraordinary}

D9= {car , championship}

Bucket B1= {speed, dream, wheels, carmaker, car, Enzo, Baracca, plaque, horse, racing, California, film, Extraordinary, championship}

Bucket B2= {machine, car}

Bucket B(B1+B2)={ speed, dream, wheels, carmaker, car, Enzo, Baracca, plaque, horse, racing, California, film, Extraordinary, championship, machine }

Fig 7: Final Bucket List

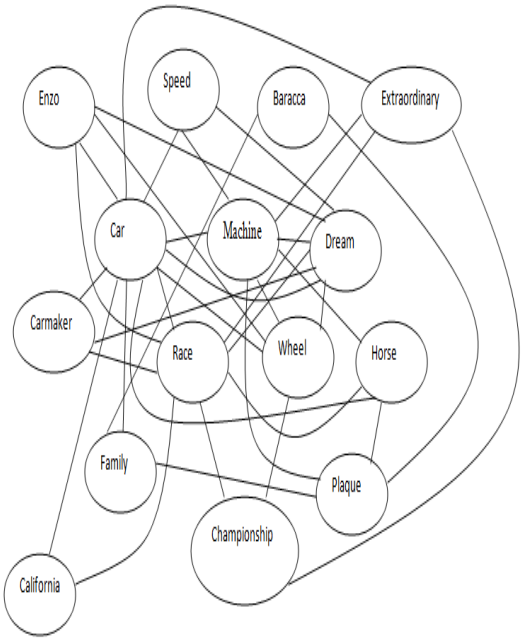

Fig 8: Graph Method Implementation

In the above given graph , number of nodes present are 15 and the number of edges are 35.To find words eligible to be keywords we divide number of edges with the number of nodes i.e.

Threshold for keywords = Number of edges/Number of nodes

Threshold for keywords = 35/15=2.33

We take threshold value equivalent to 3 because the number of edges cannot be in fraction value. Hence California and Baracca are rejected as they do not cross the threshold value and

Finally the selected keywords are ={ Speed, Dream, Wheel, Carmaker, Car, Enzo, Plaque, Horse, Race, Family, Extraordinary, Championship, Machine}.

4. RESULTS

The proposed algorithm in this paper gives a domain splitting method for keyword extraction which tries to overcome the drawbacks which were seen in the algorithms which use tf-idf method for keyword extraction. In those algorithms if a word occurs multiple times in a single paragraph but not in the overall document TF-IDF will not consider the word as keyword considering its low frequency with respect to overall document. Such words can represent an important context of document. The proposed algorithm overcomes this drawback with the help of domain splitting method where document is split into domains and keywords are extracted from each domain. If a word appears in more than half of the domains it is also considered as a keyword. Hence words are seen in each domain rather than considering their frequency with respect to the overall document. Further a graph is made of the keywords extracted to find the best optimal keywords.

5. CONCLUSION

The proposed algorithm in this paper has shown an improved method for the extraction of keywords from a set of documents. Every word is checked for its degree of relevance in each of its domains rather than checking for its relevance in the complete document. Further a graph is constructed to find the best optimal keywords from the extracted set of keywords based on their co-occurrence with one another. The graph hence gives a better structure to algorithm by relating the words found to one another. Hence this algorithm is a combination of domain splitting and graph based approach for finding relevant keywords from a document or a set of documnents. In future this algorithm can be expanded using some semantic approaches which would make it more efficient in finding keywords. Semantic approaches would be combined with statistical approaches. Word Net can be used to understand the semantics of the words and find relationships between the words use the information for efficient keyword extraction.

6. REFERENCES

[1] Information Retrieval Research**,** Jonathan Furner, School of Information and Media Studies, and David Harper, School of Computer and Mathematical Studies, The Robert Gordon University, Aberdeen, Scotland. (Eds)

[2] Important problems in information retrieval, Dagobert Soergel College of Library and Information Services University of Maryland College Park, MD 20742

[3] "Keyword extraction-a review of methods and approaches" Slobodan Beliga University of Rijeka, Department of Informatics Radmile Matejčić 2, 51 000 Rijeka, Croatia

[4] Effective Approaches For Extraction Of Keywords Jasmeen Kaur, Vishal Gupta, ME Research Scholar Computer Science & Engineering, UIET, Panjab University Chandigarh, (UT)-160014

[5] Understanding Inverse Document Frequency: On theoretical arguments for IDF, Stephen Robertson Microsoft Research 7 JJ Thomson Avenue Cambridge CB3 0FB UK

[6] Keyword Extraction using graph based approaches, R. Nagarajan, Dr. S. Anu H Nair, Dr. P. Aruna, N. Puviarasan Department of Computer Science & Engineering, Annamalai University, Tamilnadu, India

[7] Salton G, Wong A and Yang C, "A vector space model for automatic indexing", Communications of the ACM, 18(11), 613 – 620, 1975

[8] Cohen J. D., "Highlights: Language and Domain-independent Automatic Indexing Terms for Abstracting",Journal of the American Society for Information Science, 46(3): 162 – 174, 1995

[9] Mihalcea R and Tarau P, "Textrank: Bringing order into texts", In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Barcelona, Spain, 2004

[10] Jasmeen and Vishal,"Effective approaches for extraction of keywords", IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 6, November 2010 ISSN (Online): 1694-0814

[11] Hulth A., "Improved automatic keyword extraction given more linguistic knowledge", In Proceedings of theConference on Empirical Methods in Natural Language Processing (EMNLP'03), 216 – 223, Sapporo, 2003

[12] Hulth A, "Combining machine learning and natural language processing for automatic keyword extraction",PhD Thesis, Stockholm University, Faculty

of Social Sciences, Department of Computer and Systems Sciences, 2004

[13] Whitney P, Engel D and Cramer N, "Mining for surprise events within text streams". Proceedings of the NinthSIAM International Conference on Data Mining, Society for Industrial and Applied Mathematics, 617–627, 2009

[14] Salton G, Wong A and Yang C, "A vector space model for automatic indexing", Communications of the ACM, 18(11), 613 – 620, 1975

[15] I. H. Witten, G. W. Paynter, E. Frank, C. Gutwin, C. G. Nevill-Manning, "Kea: Pra-ctical Automatic Keyphrase Extraction" inProc. of the 4th ACM Conf. of the Digital Libraries, Berkeley, CA, USA, 1999.

[16] P. D. Turney, "Learning to Extract Keyphrases from Text" in Tech. Report, National Research Council of Canada, Institute for Information Technology, 1999.

[17] T. D. Nguyen, M.-Y. Kan, „Keyphrase extraction in scientific publications" in Proc. of ICADL 2007, pp. 317-326, 2007.

[18] M. Krapivin, A. Autayeu, M. Marchese, E. Blanzieri, N. Segata, "Keyphrases Extraction from Scientific Documents: Improving Machine Learning Approaches with Natural Language Processing" in Proc. of 12th Int. Conf. on Asia-Pacific Digital Libraries, ICADL 2010, Gold Coast, Australia, LNAI v.6102, pp. 102-111, 2010

[19] Y. HaCohen-Kerner, "Automatic Extraction of Keywords from Abstracts" in Proc. of 7th Int. Conf. KES 2003 (LNCS v. 2773), pp, 843-849, 2003.

[20] M. Litvak, M. Last, "Graph-based keyword extraction for single-document summarization" in ACM Workshop on Multi-source Multilingual Information Extraction and Summarization, pp.17-24, 2008.

[21] Z. Yang, J. Lei, K. Fan, Y. Lai, "Keyword extraction by entropy difference between the intrinsic and extrinsic mode" in Physica A: Statistical Mechanics and its Applications, V. 392, I. 19, pp. 4523-4531, 2013.

[22] Slobodan beliga, University of Rijeka, Department of Informatics Radmile Matejčić 2, 51 000 Rijeka, Croatia,"Keyword extraction a review of method and approaches"

[23] Y Matsuo," Keyword Extraction from a Single Document using Word Co-occurrence Statistical Information",International Journal on Artificial Intelligence Tools c World Scientific Publishing Company

[24] "Domain keyword extraction technique: A new weighting method based on frequency analysis" Rakhi Chakraborty ,Department of Computer Science & Engineering, Global Institute Of Management and Technology, Nadia, India

[25] Willett, P. (2006) The Porter stemming algorithm: then and now. Program: electronic library and information systems, 40 (3). pp. 219-223.