ISSN 2278-3091
Volume 9, No.4, July – August 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse204942020.pdf
https://doi.org/10.30534/ijatcse/2020/204942020

Combinational Semantic Approach for Efficient Data Extraction from Multiple Documents

Praveen K. Wilson [1], J. R. Jeba [2]
[1] Department of Computer Science & Information Technology, College of Engineering Trikaripur, India, praveenkwilson@gmail.com
[2] Department of Computer Applications, Noorul Islam Centre for Higher Education, India, jrjeba@rediffmail.com

ABSTRACT

As the availability of data increase in the form of web pages, there arises a challenge of effective processing of data in a timely manner. Millions of documents were added day by day in World Wide Web, and for a manual processing it will take thousands of years and hence the term automatic data extraction comes to the context. On these gigantic volumes, textual data holds a major portion and effective text processing algorithms are needed for the processing and acquisition of data. Even though a number of techniques are available for sentence extraction no one can perform well as that of a human expert. But semantic approaches can perform better than other existing approaches, since they are considering the meanings rather that its form. Here the similarity calculations are more accurate than other approaches and the level of accuracy depends on the semantic tool they have used and the efficiency of the logic used for extracting the meaning, calculating similarity etc. In this proposal we are presenting a combinational approach of statistical and semantic procedures for sentence extraction which mainly differs from the previous approaches in the use of the semantic tool ThemeSets.

Key words: Data Extraction, ThemeSets, Anaphora Resolution, Sentence Similarity, Semantic Score Calculation.

1. INTRODUCTION

Data is treated as precious and everything is decided on basis of data in the era of big data analysis and cloud computing. Each and every domain data becomes more important and these data is available in a distributed manner in various format. Some may be in the form of textual data, some may be in the format of sound some may be images etc. and much more contents are available in World Wide Web. As the availability of data increases, there arises a challenge of effective processing of these data. On these gigantic volumes, textual data holds a major portion and effective acquisition of data can be ensured if it is processed by a language expert after reading or analyzing the document. But this approach is practical only if the content is less in volume, but the actual scenario has not been matching with this condition. Millions of documents were added day by day, and for a manual processing it will take thousands of years and hence the term automatic sentence extraction more important.

Now the ball in the court of natural language processing and the area still in the lime light of researchers. Various extraction techniques were proposed by researchers and the goal is to provide most appropriate data from the large volume with respect to the input query. Based on the result provided, all these approaches can be categorized into two – extractive or abstractive. In extractive techniques it will select most suitable sentence or sentences according to the implementation algorithm. The accuracy depends on the logic behind the algorithm. Second one is abstractive methods, which gives more importance for acquiring the knowledge than selecting critical sentences. Conceptually second one is the better approach that may provide outputs as done by a language expert, but complex to implement.

Some traditional methods are based on simple mathematical calculations and have not been up to the mark on accuracy. Some may depend on domains such as news articles; medical data etc. and have limitations when we change the domain from one to other. Hence there is some space for semantic approaches, and works more precisely than the traditional approaches. In this paper we are proposing a combination of statistical and semantic approaches and is performed well than the traditional approaches.

2. BACKGROUND

Automatic text summarization is one application of data extraction in which critical and most significant sentences are extracted from a huge volume of textual data and made a summary to give as an output. In such cases the area of consideration may be single document or a collection of multiple documents. If we need to extract summary from a single document, then the challenges are comparatively

simple than it from multiple documents. In the case of multiple documents as input then redundancy, order of placing extracted sentences etc. are the new challenges we need to solve.

In almost all applications in the area of natural language processing needs to consider multiple documents as input and is especially in summarization. In case of multiple documents as input it has two extracting options. One is by giving a key word or theme as an additional input and other is summarization without giving a key word. As we told earlier, World Wide Web has a huge collection of documents with various information contents. Some may be relating to the given key word and some may not have any relation with the input key word. In case of summarization by giving a key word or theme, results significant sentences about the key word or theme. But if we are not giving a key word it will produce a generic summary that contains the significant sentences from all the input documents given. This approach has more importance in domain specific summarization and several research works are going on this area[4].

Research in sentence extraction starts in the middle of 20th century [3] and has provided significant achievements using different approaches. Initial works on key sentence extraction is based on mathematical calculations and later it extends to vast variety of areas such as fuzzy logics[10], clustering[7], ontological [12] etc. even though multiple approaches are available for sentence extraction we couldn't find an approach providing an accuracy level as given by a language expert. But semantic approaches can perform better than other existing approaches. Since they are considering the meanings domain limitations are not affected normally. But in such cases the level of accuracy depends on the semantic tool they have used and the efficiency of the logic used for extracting the meaning, calculating similarity etc.

In this proposal we are presenting a combinational approach of statistical procedures and semantic procedures. It also differs from the previous approaches in the case of the sematic tool used and the inclusion of the procedure anaphora resolution[1] computed semantically. Here we are using the features of the semantic database ThemeSets in which the lexical units are connected thematically.

3. EXISTING APPROACHES – A BRIEF NOTE

As we mentioned earlier sentence extraction is well connected with document summarization. On this aspect, the research on data extraction or specifically sentence extraction has been started in the end of 1950s. From that point a large number of approaches have been proposed by many researchers from the world based different logics, using different tools and in a variety of domain and languages. We couldn't describe all such approaches in the limited pages, but trying to present some dominant technologies starting from the traditional statistical approaches.

Sentence extraction from multiple documents has more applicability than the extraction from a single document and is more challenging. So here we are giving priority to the extraction approaches from multiple documents. Here we are discussing five dominating approaches used for sentence extraction. First one is the traditional statistical approach which is more suitable for extraction from a single document. Here we are discussing five important approaches using different technologies. We starts from the basic approach based on mathematical calculation, ie statistical approach, then feature based approach then goes to modern approaches such as graph based, machine learning and ontology based approaches.

3.1 Statistical Approach

In this approach we are using mathematical calculations for finding the key sentences from a single input or from multiple documents. It includes the initial approach proposed by Luhn in 1958[3] which is based on the assumption that important words repeated in a document.ie, here we use word frequency to find the key sentences. It then calculates probability of occurrence of a word in the document instead of finding count of occurrence of a word in the document.

$$P(w) = \frac{C(w)}{T} \quad (1)$$

Where P(w) is the inclusion probability of a word, C(w) is the count of occurrence of the word in the input document and T is total number of word in the document. Researchers made different variations in this basic approach and can be included as an additional feature of their approaches. Later the term frequency has been replaced with proportional frequency according to term frequency-inverse document frequency to improve the accuracy[13].

3.2 Feature Based Approach

In this approach we are considering different features of the sentence according to the document to find how much it is critical to be included in to the summary. This criticality is determined by analyzing the features of the sentences like its position in the document, its length, and other factors like whether it is a title sentence, whether it contains a noun or pronoun etc. according to the applied logic features under consideration are different and a weight value is assigned to every features to calculate the score of the sentence in the document. It's a commonly acceptable approach and can be included as an initial step in every approach. Some modern approaches also used these feature consideration and they use genetic algorithms to find the best combination of the features and weights[16]. General equation for calculating score of a sentence according to the features and weights is as follows:

$$Score(S_i) = \sum(S_{f1} * w_1, S_{f2} * w_2, S_{f3} * w_3, \ etc) \quad (2)$$

Where S_i is the calculated score value in the document and S_{fl} and w_1 are the corresponding sentence feature and weight. We

are taking the sum of all feature weight combination to calculate the final score of thee sentence. The number of features, type of features and corresponding weights may vary according to the logic of the algorithm. The limitation of the approach is it does take the meanings of the words or sentences into consideration.

3.3 Graph Based Approaches

Graph is a mathematical model and can have an efficient usage in the process of sentence extraction. Here each sentence can be treated as the vertices and the edges denote the relations or connectivity between the sentences. The weight associated with each edge indicates the level of bonding between the sentences. If the weight of the edge goes to zero after similarity calculations then we treat there is no such edge exists. In extraction methods, ranking of nodes may be done for selecting the critical sentences. In some cases iterative ranking algorithm is used for performing the ranking operation[19]. The algorithm used for ranking may be either syntactic or semantic[16]. According to the approach the connectivity relations between the nodes varied. If it is syntactic then the connectivity is decided by the syntactic features or statistical values. If it is semantic the relations will be decided by the semantic tool used. The efficiency of such approaches depends on the accuracy of semantic similarity calculations which may add more weights to the edges if the vertices are semantically similar.

3.4 Machine Learning Approach

With the usage of machine learning approaches, the term corpus is also come into consideration. Corpus is a huge data set used for giving sufficient training for the system for selecting critical sentences. The learning process may be supervised or unsupervised, but in majority cases supervised learning preferred. From the training set and corresponding selected sentences, the model may be able generate a procedure for selecting the key sentences and this training process continues with more and more training data and corresponding selected sentences. After successful training the model may be able to get key sentences from any of the given input documents. On the process of selection it works as a classifier using any of the classification algorithms. It classifies the input sentences into two categories – one is the set of key informative sentences and a set of other unwanted sentences. In the initial proposals, the Naïve Bayes classifier is used for the score calculation of sentences which leads to the classification[21][23]. The equation used for score calculation is given below:

$$P(y|x_1, x_2, \dots, x_n) = \frac{P(y) * \prod_{i=1}^{n} P(x_i|y)}{P(x_1) * P(x_2) * \dots * P(x_n)} \qquad (3)$$

Where y denotes a specific sentence for score calculation and x_1,x_2,..x_n are the disjoint features. This approach has one major limitation that it has much dependence on the corpus hence the performance may vary according to the input domain.

3.5 Ontology Based Approach

This approach has some deviations from the traditional concepts of data extraction. It is commonly used for extracting data from a specific domain. The main idea behind this approach is the knowledge base or ontology and generally this knowledgebase is from a specific domain. This ontology concept helps to perform similarity checking even with a domain specific terms if the knowledge base is from same domain. Later this approach extends to fuzzy ontology which helps to apply the ontological concept with the input query[22]. The major limitation of this approach is its domain dependency. We cannot extract sentences based on a general query by using this approach and the accuracy depends on the result of how perfectly the ontology is created.

Here we have seen some of the existing approaches based on different principles used for the extraction of data or specifically textual data from a huge volume of inputs. These are small drops in count among the ocean of techniques developed by the researchers all over the world, but these are dominant among the others based on technology. Even though a large number of approaches are available, no one can provide a result as perfect as given by a language expert after reading and analyzing.

4. PROPOSED ALGORITHM

Among the number of approaches available for sentence extraction, semantic methodologies have some importance, since it considers the meanings also in the process of selecting critical sentences. Here we are proposing a combination of syntactic and semantic approach which uses some syntactic or statistical scores and a semantic score calculated using a semantic tool ThemeSet[1] for finding the critical sentences to be given as the output. In this combinational approach we are giving one by fourth weightage is given for the syntactic values and three by third weightage is given for semantically calculated values.

In syntactic score calculation we are considering syntactic features like frequency, sentence position, sentence type, presence of proper nouns etc. For semantic score calculations, we are using two semantic similarity checking methods, the famous normalized Google distance and a new approach thematic similarity which can be calculated using the semantic tool ThemeSet. In thematic similarity calculation the verbs are considered to find how much the statements are adjacent. Also as an initial step we are performing anaphora resolution to make similarity checking more accurate. Here also we are applying thematic similarity.

4.1 Methodology

As mentioned earlier, the methodology starts from statistical calculation followed by semantic calculations. Among that anaphora resolution is included for getting result more accurately. The entire procedure we have used for semantic data extraction is as given below:

A. Preprocessing

Here we are converting the entire input documents to the format suitable for further processing. The entire textual content is divided into sentences and gave a unique identifier for identifying each sentence. Also we are removing all contents which are not giving any valid information for score calculation such as "a", "an", "the" etc. But we are retaining all pronouns and noun referrals since we need to resolve it in the anaphora resolution phase.

B. POS Tagging

In this step we are using a tool parts of speech tagger. Using a POS tagger we are able to find the part-of-speech of each tokens or words in a sentence. Here we are using the famous NLTK toolkit for part-of-speech tagging.

C. Lemmatization

Here we are converting all tokens excluding nouns and noun referral from its inflectional or derivational forms to the base form or dictionary form called lemma. It is done for making similarity checking more accurate. For this purpose we are using Wordnet Lemmatizer with NLTK.

D. Noun Referral Resolving

In this step we are replacing all noun referrals including pronouns with corresponding noun phrase. This leads to an increase in the number of similar sentences with the input query and creating a large pool for selecting critical sentences. Here also we are following a semantic approach by using the semantic tool Themesets[1].

E. Statistical Score Calculation

Here we are using the statistical and feature parameters for calculating statistical score. Weights are associated with each parameter values according to the importance in score calculation. The main parameters we are using for score calculation are maximum likelihood value, sentence position in the document, and presence of title word. Statistical score can be calculated by the following equation:

$$\text{Stat\_S}(S_i) = MLV * (Pos(S_i) * w_p + T(S_i) * w_t) \quad (4)$$

Where MLV is the maximum likelihood value between zero and one which denotes the syntactic similarity of the target sentence S_i with the input query and w_p and w_t are weights associated with the positional value and title status of the target sentence.

F. Semantic Score Calculation

We are calculating the semantic score by applying two semantic similarity measuring procedures – one for calculating similarity based on normalized google distance and other based on thematic similarity. In this combination we are giving one by third weightage for score according to normalized google distance method and two by third for thematic approach.

$$Sem\_S(S_i) = NGS(S_i) + 2 * Theme(S_i) \quad (5)$$

Where $NGS(S_i)$ is the similarity value calculated according to normalized google distance and $Theme(S_i)$ is the similarity value calculated according to Themesets for the sentence S_i. For calculating NGS value we are considering word by word similarity according to the search hits and hence for the entire sentence.

$$Dist(A,B) = \frac{\max\{\log h(A), \log h(B)\} - \log h(A,B)}{\log N - \min\{\log h(A), \log h(B)\}} \quad (6)$$

Where A and B are the lexical units and $h()$ function denotes the search hits returned by google for the occurrence of corresponding terms. From the distance value between the terms, we are calculating the NGS value using the below equation:

$$NGS(S_i) = \frac{\sum_{A \in S_Q} \sum_{B \in S_i} Dist(A,B)}{|S_Q| . |S_i|} \quad (7)$$

S_i and S_Q are the target sentence and the search query respectively. Thematic similarity value can be calculated by considering the bond dependency value and the relations between the lexemes. According to [1]ThemeSets have the form:

$$TS(W) = \{W_1 R_1 [B_1], W_2 R_2 [B_2] \dots \dots \dots\} \quad (8)$$

W_1,W_2 are the lexemes in the Themsets corresponding to the word W and R is the connecting relations with bond dependency value B.

$$Theme(S_i) = \frac{|lcs(S_Q, S_i)| * \sum_{A \in S_Q} \sum_{B \in S_i} B_{AB} * R_w}{\min\{|S_Q|, |S_i|\}} \quad (9)$$

Where B_{AB} is the bond dependency value between the lexemes A and B and R_w is weightage given for the connecting relation.

G. Critical Sentence Selection

Here we are assigning score to every sentence in the input document according to the statistical and semantic values.

$$S_{final}(S_i) = Stat_{S(S_i)} + NGS(S_i) + 2 * Theme(S_i) \quad (10)$$

We are classifying the input sentences into group of selected sentences and a group of rejected sentences. Also we are assuming two threshold values T_1 and T_2 as decision parameters for the inclusion and rejection of sentences.

If the S_{final} value between the target sentence and input query is greater than T_1 and corresponding value with already selected sentences is less than T_2, then we include the target sentence into the group of selected sentences. If the S_{final} value between the target sentence and input query is greater than T_1 and corresponding value with already selected sentences is also greater than T_2, then we reject the target sentence to avoid redundancy. However the sentence in the group of selected sentences by which the target is rejected got an improvement in score value by a factor f denoting it needs high priority.

After completing the procedure with all input documents we select sentences with high score as critical sentences to a level according to our requirement. The accuracy of selection lies in similarity score calculation and hence the perfectness of the ThemeSets.

4. RESULT AND DISCUSSION

The analysis was done in python programming language, and the NLTK has been used for the preprocessing of the input documents. The main challenge we have faced is the lack of availability of a well-defined ThemeSet in a form suitable for the calculations. We have overcome the issue by using a prototype of the ThemeSet created by including the lexemes available in the input documents. For the evaluation we have used a collection of 44 documents with an overall sentence count of 2280 from various domains. We have given the documents as inputs to the proposed model and evaluate the selected sentences got from the model as output after performing the entire procedure. The same input has been given to two more traditional approaches and corresponding performance has been compared with the result of proposed method. The correctness of the retrieved data has been evaluated by language experts and performance of the proposed methodology was compared with the traditional approaches by using quality measuring parameters.

The common quality measuring parameters that used for evaluating natural language processing approaches are precision and recall. It evaluates the approaches on the basis of correctness and completeness of the resulting data values. Here precision value denotes the correctness of the selected sentences and recall value denotes the measure of completeness of the result given by the corresponding model. For performance comparison we are also calculating the precision and recall values of two traditional approaches- statistical method and feature based method by giving the same input documents for data extraction. The parameter values can be calculated by the below equations:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (11)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (12)$$

$$\text{F1} - \text{Score} = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (13)$$

Where true positive (TP) indicates the number of sentences correctly included in the group of selected sentences and true negative (TN) indicates the number of sentences correctly included in the group of rejected sentences. False positive (FP) denotes the number of sentences incorrectly selected as critical sentence and false negative (FN) denotes number of sentences incorrectly rejected. F1-Score or F-Measure is the weighted average of precision and recall.

After analyzing the output sentences given by the three approaches, we have calculated the positive and negative values in both true and false cases. Table 1 shows the calculated positive and negative values according to the classification of sentences and Table 2 gives the corresponding quality parameter values calculated using the above equations for the three approaches.

Table 1: Comparison of Classification Parameters

Methodologies	TP	FP	FN	TN
Statistical	46	42	40	2152
Feature Based	48	26	38	2168
Combinational Semantic	70	15	16	2179

The comparison of proposed methodology- combinational sematic with traditional approaches in terms of quality measuring parameters is given below:

Table 2: Comparison Performance Values

Methodologies	Precision	Recall	F1-Score
Statistical	52	54	53
Feature Based	65	56	60
Combinational Semantic	82	81	82

A graphical representation for the comparison of quality measuring parameters is shown below. Here the parameter values are presented as percentage except F1-Score.

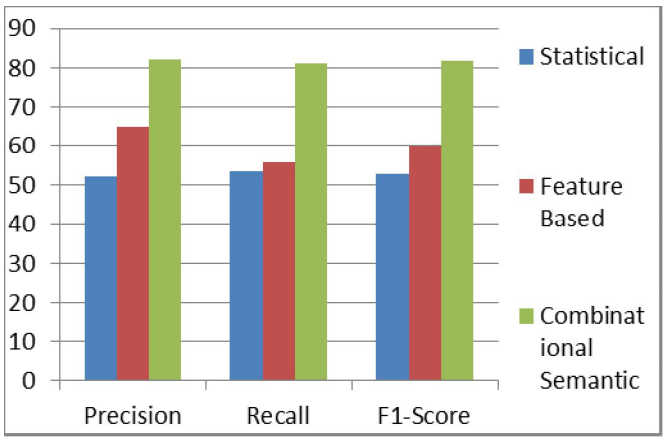

Figure 1: Quality Measuring Parameters

From the tables and graph we can see a much better performance of our proposed combinational semantic approach than the traditional approaches such as statistical based and feature based methodologies. It provides a minimum improvement of 17% in correctness, 25% in completeness and 22% in F1-Score value.

5. CONCLUSION

The proposed methodology based on the combination of statistical and semantic approach performs well in the process of sentence extraction despite of the input domain. It leads to the assumption that semantic approaches can give output nearly up to the level or better than a language expert can do. Hence doors are open for researchers for getting a hundred percentage results. The limitation of our approach lies in the

absence of a well-structured ThemeSet in a form suitable to perform calculations. In future the proposed methodology can be improved by increasing the efficiency of the ThemeSet by sharpening the bond dependency value and implementing more connectivity relations with the help of a large corpus.

REFERENCES

1. Praveen K Wilson, J R Jeba, **Anaphora Resolution Using ThemeSets**, *International Journal of Innovative Technology and Exploring Engineering (IJITEE)* Volume 08 Issue 10. August 2019.
2. Dixit Rucha S., Apte S. S., **Improvement Of Text Summarization Using Fuzzy Logic Based Method**, *IOSR Journal Of Computer Engineering (IOSRJCE) ISSN: 2278-0661, ISBN: 2278-8727,* Vol. 5, Issue 6, PP 05-10, 2012.
3. Luhn, H.P., **The automatic creation of literature abstracts**. *IBM J. Res. Dev., 2:* 159-165. DOI: 10.1147/rd.22.0159, 1958.
4. Nenkova, A. and K. McKeown,. **Automatic summarization**. *Foundat. Trends Inform. Retrieval*, 5: 103-233 2011.
5. Sarkar Kamal, Nasipuri Mita, Ghose Suranjan, **Using Machine Learning for Medical Document Summarization**, *International Journal of Database Theory and Application*, 2011.
6. Plaza Laura, Díaz Alberto and Gervás Pablo, **A semantic graphbased approach to biomedical summarisation**, *Artificial Intelligence in Medicine* 53, 2011. https://doi.org/10.1016/j.artmed.2011.06.005
7. Khanapure V.M, Prof. Chirchi V.R **Multi-document Summarization Based on Cluster**, *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering* Vol. 3, Issue 4, 2014.
8. S. S. Sonawane, P. A. Kulkarni, **Graph based Representation and Analysis of Text Document: A Survey of Techniques**, in *Int. Jour. Of Computer Applications* 96(19):1-8, 2014.
9. F. Boudin, **A comparison of centrality measures for graph-based keyphrase extraction**, in *Int. Joint Conf. on Natural Language Processing (IJCNLP)*, pp. 834-838, 2013.
10. Makrehchi, M., Kamel, M.: **A fuzzy set approach to extracting keywords from abstracts**. *IEEE Int. Conf. Fuzzy Inf. 2*, 528–532 (2004).
11. C. Huang, Y. Tian, Z. Zhou, C.X. Ling, T. Huang **Keyphrase extraction using semantic networks structure analysis** in *IEEE Int. Conf. on Data Mining*, pp.275-284, 2006.
12. Ramezani Majid, Feizi-Derakhshi Mohammad-Reza, **OntologyBased Automatic Text Summarization Using FarsNet**, *ACSIJ Advances in Computer Science: an International Journal*, 2015, Vol. 4, Issue 2, No.14.
13. Filatova, E. and V. Hatzivassiloglou,. **A formal model for information selection in multi-sentence text extraction**. *Proceedings of the 20th International Conference on Computational Linguistics*, (CCL' 04), ACM, pp: 397-4032004.
14. Y. Gong and X. Liu, **Generic text summarization using relevance measure and latent semantic analysis**, *Proc. SIGIR*, pp. 19-25 ,2001. https://doi.org/10.1145/383952.383955
15. Ling Zheng, Hui Gui and Feng Li, **Optimized Data Preprocessing Technology For Web Log Mining**, *IEEE International Conference On Computer Design and Applications(ICCDA)*, pp. VI-19-VI-21,2010.
16. Arun Krishna Chitturi, Saravanakumar Kandasamy, **Survey on Abstractive Text Summarization using various approaches**, *International Journal of Advanced Trends in Computer Science and Engineering, Volume 08 Issue 06*,2019
17. Bossard, A. and C. Rodrigues, **Combining a multidocument update summarization system**–*CBSEAS*– *with a genetic algorithm. Combinat. Intell. Methods Applic.*, 8: 71-87. 2011.
18. JING Chang-bin and Chen Li, **Web Log Data Preprocessing Based On Collaborative Filtering**, *IEEE 2nd International Workshop On Education Technology and Computer Science*, pp.118-121, 2010.
19. Borhan Samei and Marzieh Eshtiagh **Multi-Document Summarization Using Graph-Based Iterative Ranking Algorithms and Information Theoretical Distortion Measures**, *Proceedings of the Twenty-Seventh International Florida Artificial Intelligence Research Society Conference*, 2014
20. Koehn, Philipp and Hieu Hoang. **Factored translation models**. *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, ACL, pages 868–876, 2007.
21. Sarkar Kamal, Nasipuri Mita, Ghose Suranjan, **Using Machine Learning for Medical Document Summarization**, *International Journal of Database Theory and Application*, 2011.
22. Lee, C.S., Z.W. Jian and L.K. Huang, **A fuzzy ontology and its application to news summarization** *IEEE Trans. Syst., Man Cybernet. Part B: Cybernet*, 35: 859-880, 2005. https://doi.org/10.1109/TSMCB.2005.845032
23. M.Govindarajan, **A Comparative Analysis of Ensemble Classifiers for Text Categorization** *International Journal of Advanced Trends in Computer Science and Engineering*, Volume 09 Issue 01, 2020.