International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-1, November 2019

Automatic Summarization of Textual Document

Faiyaz Ahmad, Yassar, Amreen Ahmad

***Abstract*: In today world, there is a huge amount of information is growing every day on the internet and from many other sources and there is lots of textual information in it. To find out the relevant information from this large amount of data, we need an automatic mechanism that will extract the useful data. Such automatic systems are automatic summarization systems. They categorized into extractive and abstractive summarization system. Extractive summarization systems select the important sentences directly from the large document and put into summary whereas abstractive methods understand semantic meaning of the document by linguistic method to interpret and examine the text.**

In the purposed method, a statistical approach is used where multiple criterions or features are discussed to calculate the score for every sentence and then SIR (Susceptible Infected Recovered) model is used to compute the dynamic weight for every feature. After dynamic weight computation, weighted TOPSIS (The Technique for Order of Preference by Similarity to Ideal Solution) is used for multi-criterion analysis and aggregation. This method is fully implemented and integrated for automated textual document summarization system.

***Keywords*: Extractive, SIR, Weighted TOPSIS, Single document, Multi-document, Generic summaries, Precision and Recall, Bigrams and skip-gram**

I. INTRODUCTION

Automatic Summarization of textual document is a method of creating a small and accurate summary of a large document that will contain relevant and useful information of large document.

The huge amount of information is growing every day on the internet and from many other resources and there is lots of textual information in it [1]. Most of the data is unstructured and it is very hard to find out the relevant information from this large amount of unstructured data. Summarization systems are used to extract useful information from a large amount of data. There are many reasons for generating the summaries of a large document. One example we can easily see in the daily life of newspaper. They use heading as a summary of the paragraph but there are many more reasons why summarization systems are much important in today's life.

Revised Manuscript Received on November 30, 2019.
***** Correspondence Author
Faiyaz Ahmad*, Dept. of Computer Engineering, Jamia Millia Islamia, New Delhi -India. Email: ahmad.faiyaz@gmail.com
Yassar, Dept. of Computer Engineering, Jamia Millia Islamia , New Delhi India. Email: aliyaser78691@gmail.com
Amreen Ahmad, Dept. of Computer Engineering, Jamia Millia Islamia, New Delhi -India. Email: amreen.ahmad10@gmail.com

There are different aspects to understand the summarization system that can be classified according to system input type (multi-document or single document), Output type that can be extractive or abstractive and Purpose (query-based and generic) [2].

Single document summarization system: In Single-document summarization system, the summarization system generates the summary of only one single document [3]. In early there were many summarization systems that deal with only single document but in today world multi-document summarization system is also used.

Multi-document summarization system: Multi-document summarization system takes multiple documents as input and produces the summary by concatenating them. Here one assumption is made that these multiple documents are often related to the same topic or same discussion [2]. Today this type of summarization system is used very frequently because lots of information is generating on the internet and much information are related to each other. Early days when there was no much information we were using single-document summarization systems.

Generic summaries: In a generic summarization system, the system generates the summary of textual documents without knowing its topic or domain. The generic summarization system views all documents as homogenous texts. They do not make any assumptions about the domain or topic of document [4].

Query-based summaries: Query-based summarization system generates the summary according to the query posed by user. These queries are simple English like language queries or some keywords of language that can be according to a specific subject. A snippet produced by search engines is an example of a query-based application.

Extractive summarization system: Extractive summarization systems directly select the important sentences from the large document based on their importance and put into summary. In early researches, most of the summarization systems were the extractive type summarization systems [2]. The extractive summarization system is an easy approach that's why most successful text summarization methods are extractive.

Abstractive summarization system: Abstractive summarization systems are inspired by human thinking and his summary generation creativity. How human makes the summary of long discussion or movies the same idea is incorporated into abstractive summarization systems. Abstractive summaries are more challenging than extractive summaries because they need a linguistic method to understand the meaning of source document. One thing need to consider that Abstractive text summarization may generate entirely new sentences or phrase that may not be the part of the original document [3].

II. RELATED WORK

Erkan et al. proposed some graph centric methods. In graph centric method, graph nodes represent the sentences of the document and edges come between the sentences when they have some semantic similarity between them. They used LexRank idea which was the multi-document summarization system idea [5]. After network generation, they use a random walk technique on graph to extract useful and relevant sentences or nodes.

Baralis et al. (2013) proposed the GRAPHSUM method that is used as a generic summarization system and it was based on the graph in which they use association rules between the multiple terms of graph [6].

ZHANG Pei-ying et al. [7], purposed clustering and extraction approach for ATS. In this approach, they followed three steps. First they made the clusters of sentences based on semantic similarity between sentences after that they compute the accumulated sentence similarity using multi-features combination method and lastly, they extracted some sentence form the clusters using some extraction rules. Sentences are clustered based on similarity. Similarity computed by many different methods. They use the k-mean method for clusters generation [8].

Suanmali, L., Binwahlan et al. [3] proposed a fuzzy-based summarization system. In their method, they consider many different features such as length of sentences, the position of sentences and much more and each feature was the input to the fuzzy system [4]. They made some rules and these all rules are stored in the knowledge-based of the fuzzy system Then a fuzzy inference system sees the rules to extract useful and relevant sentences from the system [1].

Kruengkrai, C., and Jaruskulchai, C. (2015, October) [9] proposed a graph-based technique for extract the useful and important sentences from the document for summary generation. Their approach takes both global and local property of sentences. The global property can be looked at as the semantic similarity between sentences and local property as worked on the words of each sentence [10].

Some researchers have been used Machine learning-based approaches for their model. Machine learning models learn from the data. Machine learning models are categorized into supervised learning, unsupervised learning and reinforcement learning algorithms. In this type of model basically, documents are treated as independent variables and human-generated summaries for the text treated as labels for that document. The supervised model takes a new document and checks the similarity with other documents that are in train data already and it produces the summary with the highest matched document label. A large amount of labeled data is required in supervised learning algorithms for the learning phase of the model. Many machine learning models that can be used towards the classification or in summary generation like Naïve Bayes [11], logistic regression [12], support vector machine (SVM) [13] classifier and Decision trees [14] etc are some important supervised algorithms. In the unsupervised learning algorithm, there is no need for any training data at first to train the model. These methods generate the summary of the target document directly. These algorithms try to find hidden patterns or structures in the unlabelled data. These types of systems apply some rules to extract important and relevant sentences from the document for summary generation. Hidden Markov Model [15] Association rules and Clustering [16] etc are some very good algorithms that are the unsupervised learning algorithms. On the other hand, Genetic algorithms (GA) [17] and fuzzy inference systems are also the types of machine learning approaches. Genetic algorithms also come in machine learning algorithms where they solve the optimization problems. On the other hand, the fuzzy system uses a rule-based system to extract useful sentences from the document. Some machine learning algorithms are semi-supervised. For these types of algorithms, labeled and unlabeled data required to make a mapping function that will work as a classifier.

III. PROPOSED WORK

A. Proposed Architecture

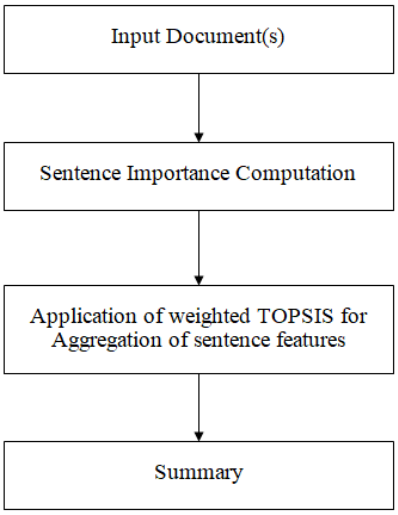

Fig. 1. Proposed Architecture

B. Input Document(s)

A Single document and multi-document is used as input for the summarization system. This multi-document is often related to the same topic or discussion.

C. Sentence Importance Computation

Importance of the sentence is computed based on some features [2]. Some past researchers have identified some features that can be very useful toward the summary generation. Although in this paper, some extra features are discussed that are not used in past research for summary generation. These features give the score for sentences. These scores can be used to extract the important sentence.

- **Sentences length criterion (f1):** In this step, length of each sentence of the document(s) is calculated as-

$$\text{.Len\_scre}(S_i) = \sum \text{words} \quad (1)$$

Where words $\in$ Si

- **Sentence position criterion (f2)**: In this step, the position score of each sentence is calculated based on their position in the document. Some researchers favored those sentences that are at the beginning of the document than those sentences that are coming at the end of the document [2].

$$\text{sent\_pstn\_scre}(Si) = m/\text{pos}(S_i) \quad (2)$$

- **Sentence connectivity criterion (f3):** In this step, the sentences connectivity score or cosine similarity score between sentences are computed that will tell how sentences are semantically related to each other. If some sentence is semantically related to many other sentences then it assumed that this sentence is important in the document and can give a high score to this sentence [2]. Equation (3) used to calculate the semantic score between two sentences.

$$Similarity(A,B) = \frac{\sum_{i=1}^{f} A_i * B_i}{\sqrt{\sum_{i=1}^{f} A_i^{\,2}} * \sqrt{\sum_{i=1}^{f} B_i^{\,2}}} \quad (3)$$

Where A is a column vector of some sentence S_i and B is a column vector of another sentence S_j. We compute the similarity of sentence Si to all other sentences of document and for final similarity score; we can add all similarity score of S_i.

- **Weighted word frequency criterion (f4):** After tokenization (split sentences into words) of each sentence the frequency of each word is calculated. After calculating the frequency of each word the weighted word frequency of each word is calculated by just dividing word frequency of a word by the frequency of the most occurring word.

$$freq(w_i) = count(w_i) \quad (4)$$

$$whted\_wrd\_freq(w_i) = freq(w_i)/max\{freq(w)\} \quad (5)$$

where $w_i \in$ Document.

For sentence score calculation add all weighted word frequency of all words which are coming in sentence.

$$whted\_wrd\_\ freq\_scre(S_i) = \sum whted\_wrd\_freq(w_s) \quad (6)$$

where $w_s \in$ Si

- **Degree centrality criterion (f5):** The degree centrality measurement of each sentence is just the number of edges which is connected to this sentence. This is the simplest centrality measure. The more central or more important sentences have max degree.

$$degree\_scre(S_i) = count(e_i) \quad (7)$$

where all e_i connected to S_i

- **Closeness centrality criterion (f6):** Closeness centrality calculates the distance of each sentence from some sentence S_i to calculate its closeness centrality score and sum all distances. Final closeness centrality score of some sentence Si calculated as (8).

$$c(v_i) = \frac{1}{\sum_j d(v_i, v_j)} \quad (8)$$

- **Betweenness centrality criterion (f7):** The betweenness centrality score for a vertex v_i can be calculated as how many shortest paths between all pairs of vertices that include vertex v_i. betweenness centrality score computed using (9). Where η_{jk} denote the total shortest path between node j and k.

$$b(v_i) = \sum_{j \neq i} \sum_{k \neq i, k > j} \frac{\eta_{jk}(v_i)}{\eta_{jk}} \quad (9)$$

D. Application of weighted TOPSIS for aggregation of sentence features

Weighted TOPSIS is used to aggregate and analysis multiple feature. This method uses the weights for each feature to compute the final sentence score of the document. Basically all features are not equally important. Same features are more important than other features. That's way SIR model is used that computes the weights dynamically based on features importance. Weights of the features will depend on the context of the document and they change when text data changes. If a dataset is changed their features weights will change automatically. SIR model tells the most influential node in the sentence network. In the SIR model, each node (in this project every sentence will be the node or vertex) has three states: (i) Susceptible S(t) states that tells about those nodes which are not infected yet by other nodes. (ii) Infected I(t) states that represent those nodes which have been influenced by other nodes. (iii) Recovered R(t) represents those nodes that were affected but now they are not affected. There is an infected rate α which tells the rate of effected nodes by others. There is also a recovered rate which tells recovered rate of nodes which denoted by β. in this project, values of α and β is selected that gives the highest similarity of the document to the referenced summary. There is an F (t) value for every node which is the summation of the infected nodes and removed nodes over the time t. based on this f(t) value most influenced node checked. This is computed by iterate over the nodes' network. Based on these f(t) values, features weights can be computed [18]. In weighted TOPSIS a features score matrix is constructed which contains all features score for every sentence of the document and then this matrix is used to find out the important sentences to form the summary. In this matrix, all columns will be the features and row will be the sentences that have to be extracted. The TOPSIS method is easy and simple to implement and this method have the ability to work on unlimited number of alternatives and unlimited number of criterion to make the decision. That's why this method is preferable over other decision-making methods [18].

Example: 4 features and 4 sentences are used in this example. A decision matrix is constructed as features score matrix (Table I) that contains sentences feature score. Weighted TOPSIS uses this matrix to perform all operations and to find out the relative closeness score for all sentences.

Table II Features Score Matrix

Sent	*f1*	*f2*	*f3*	*f4*
S1	6	7	8	6
S2	8	7	8	7
S3	7	9	9	8
S4	9	6	8	9

After features score matrix construction, normalize this matrix using (10) and Table II constructed.

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}} \quad (10)$$

Table II Normalized Decision Matrix

Sent	*f1*	*f2*	*f3*	*f4*
S1	0.40	0.48	0.48	0.40
S2	0.53	0.48	0.48	0.46
S3	0.46	0.61	0.54	0.53
S4	0.59	0.41	0.48	0.59

There is a weight for every feature which was computed dynamically by the SIR model.

Now every value of the normalized matrix is multiplied with dynamic weight of the feature using (11) and let assume w = (0.1, 0.4, 0.3, 0.3)

$$v_{ij} = w_{ij} * r_{ij} \quad (11)$$

Table III Weighted Decision Matrix

Sent	*f1*	*f2*	*f3*	*f4*
S1	0.040	0.192	0.144	0.080
S2	0.053	0.192	0.144	0.092
S3	0.046	0.244	0.162	0.106
S4	0.059	0.164	0.144	0.118

Now positive ideal solution A^+ and negative ideal solution A^- is computed according to weighted decision matrix using (12-13) and this A^+ and A^- are used to compute the separation distances between sentences using (14-15).

$$A^+ = \{v1^+, v2^+, \ldots, vn^+\} \quad (12)$$
$$where\ v_j{}^+ = \{(maxi(v_{ij})\ if\ j \epsilon J); (mini(v_{ij}) if\ j \epsilon j')\}$$

$$A^- = \{v1^-, v2^-, \ldots, vn^-\} \quad (13)$$
$$where\ v_j{}^- = \{(mini(v_{ij})\ if\ j \epsilon J); (maxi(v_{ij}) if\ j \epsilon j')\}$$

$$S^+ = \sqrt{\sum_{j=1}^{n} (v_j{}^+ - v_{ij})^2} \quad (14)$$

$$S^- = \sqrt{\sum_{j=1}^{n} (v_j{}^- - v_{ij})^2} \quad (15)$$

Where i = 1,2,……,m

So separation distances between sentences computed as:

$S^+ = \{0.058, 0.057, 0.029, 0.090\}$

$S^- = \{0.047, 0.040, 0.083, 0.019\}$

After separation of distances calculation relative closeness computed using (16).

$$C_i = \frac{S_i^-}{S_i^+ + S_i^-} \quad (16)$$

So final relative closeness of the sentences is computed as:

$C_i = \{0.45; 0.41; 0.74; 0.17\}$

E. Summary Formation

Now relative closeness score for every sentence has been computed. Now these relative scores can be used to rank the sentences. Rank to the sentences is given based on their relative closeness score. For a maximum relative closeness score of a sentence a highest rank is given to this sentence and second maximum closeness score of a sentence, second rank is given to this sentence and so on. After giving the rank to every sentence, extract the top rank sentences for summary formation. Number of sentences in the summary will be user dependent. Now these extracted sentences will be the final computed summary of the document by summarization system. This computed summary also said to be a system summary.

IV. EVALUATION

A. Dataset

Both single document (DUC2002) and multi-document (DUC2004) are used as input to the summarization system showing in Table IV.

Table IIII Dataset

Dataset	*Clusters*	*Summary*
DUC02	59	200
DUC04	50	665Bytes

The National Institute of Standards and Technology (NIST) provide the Document Understanding Conference (DUC2002) dataset for summarization. This dataset contains approx 59 text document about newswire/paper stories. This dataset also provides the 200 gold summaries for every document. This dataset provides the single document for each topic.

DUC2004 dataset also contains the documents about newswire/paper stories. This is a multi document dataset. The dataset contains 50 clusters of the document every cluster contain multiple document for same the topic. Size of this dataset is 665bytes. This dataset also provides the gold summaries for every document that can be used to check the performance of the proposed summarization system.

B. Evaluation Metrics

To evaluate the performance of summarization systems Precision and Recall is used as an evaluation parameter because these two parameters are very useful if some NLP work is done. Let's discussed what is Precision and Recall and how they are used in the field of natural language processing.

- **Recall:** Already human-generated summaries are available in the dataset for every document. This summary is said to be a gold summary or referenced summary. In this project, we compute the summary for each document by the summarization system. This summary is said to be a system summary or computed summary. So basically Recall is nothing but how much of the reference summary or gold summary is captured by the system summary.

$$Recall = \frac{number\ of\ overlapping\ words}{total\ words\ in\ system\ summary} \quad (17)$$

C. Baseline Methods

In this project Extractive summarization approach is used and all results are computed on DUC2002 and the DUC2004 dataset. The computed results by the proposed method are compared with other methods. These results are taken from the Table V and Table VI [19].

- **Random:** Sentences are selected randomly from the document to form the summary.
- **Centroid:** For summary formation, the different features of the sentence are used. Similarity is computed between candidate sentences and first sentences to extract the useful sentences.

- **NMF:** A non-negative matrix factorization is used to create a term by sentence matrix on the document and important sentences are extracted using the weighted value.
- **DUC best:** Best participant of both DUC2002 and DUC2004 conference are represented in it.
- **OCDSum:** This technique based on the differential evolution algorithm concept.
- **LSA:** A dimensionality reduction technique (singular value decomposition (SVD)) is used in which key sentences of the document are identified using computed weighted value.
- **WCS:** To form the summary of multi-documents, computed single documents ranks are combined.

D. Performance Results

Performance of the system is evaluated on recall and bi-grams and skip-grams are used because bi-grams and skip-grams are more informative about the performance of the model than unigrams. Fig.2. and Fig.3. shows the performance of purposed the method on bigram and skip-gram. Both results are computed on DUC2002 (Single document Dataset).

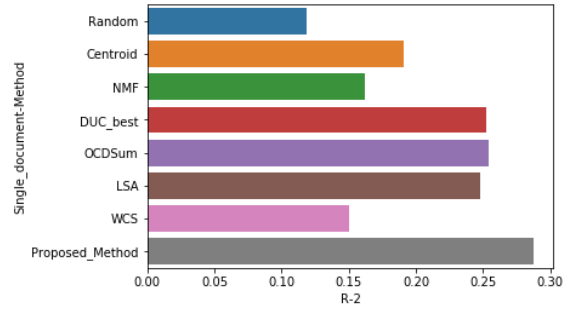

Fig. 2. Results on DUC02 (R-2)

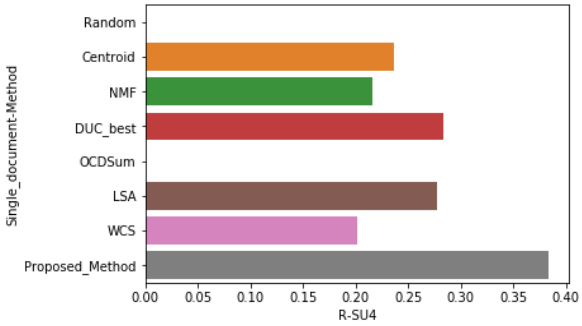

Fig. 3. Results on DUC02 (R-SU4)

Fig.4. and Fig.5. shows the results for DUC2004 (multi-document Dataset). Recall is used as a performance evaluation parameter.

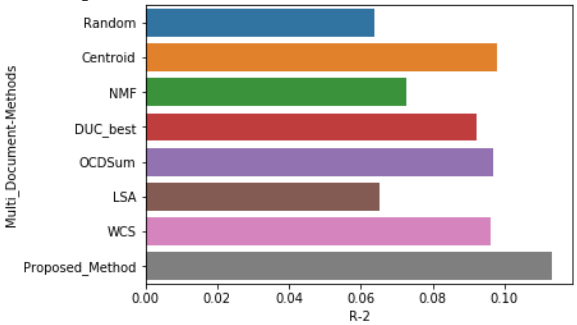

Fig. 4. Results on DUC04v (R-2)

Table V and Table VI shows the results in tabular form that are computed by the summarization system.

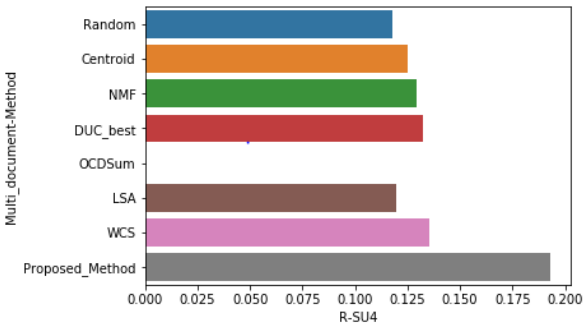

Fig. 5. Reults on DUC04 (R-SU4)

Table V Performance Results on DUC02

System	R-2	R-SU4
Random	0.1196	-
Centroid	0.1918	0.2363
NMF	0.1628	0.2169
DUC Best	0.2523	0.2841
OCDSUM	0.2548	-
LSA	0.2484	0.2789
WCS	0.1502	0.2023
Proposed Method	0.2882	0.3841

Table VI Performance Results on DUC04

System	R-2	R-SU4
Random	0.0639	0.1177
Centroid	0.0981	0.1251
NMF	0.0726	0.1291
DUC Best	0.0922	0.1323
OCDSUM	0.0969	-
LSA	0.0654	0.1194
WCS	0.0961	0.1353
Proposed Method	0.1134	0.1933

V. CONCLUSION AND FUTURE WORK

In the proposed method, a statistical approach is used where multiple features are discussed to extract the relevant and important sentences from the document. Some features already discussed in past research but in this paper some new features are discussed. These new features compute the centrality score of sentences and these features are not used in summarization systems till yet. For aggregation and analysis to multiple features, a weighted TOPSIS is used. TOPSIS method is simple and easy to implement and this method has the ability to work on unlimited number of alternatives and unlimited number of criterion to make the decision. That's why this method is preferable over other decision-making methods.

In this project, English documents are used as input which was a monolingual document and all results are computed on these documents. Although this proposed method is language independent. So this can be extended for multilingual document where a document can be use that will contain multiple languages or all multiple documents will be in different languages.

This work also can be extended as add an external image in the summary from the internet if there is no image in the document because images gives a better understanding than text in short time.

REFERENCES

1. Kyoomarsi Kyoomarsi, F., Khosravi, H., Eslami, E., Dehkordy, P. K., & Tajoddin, A. (2008, May). "Optimizing Text Summarization Based on Fuzzy Logic" *In Computer and Information Science,* 2008. ICIS 08. Seventh IEEE/ACIS International Conference on (pp. 347-352). IEEE.
2. Mehdi-Belguith, Imen-Touati, Mohamed-H´edi Maˆaloul, & Iskandar-Keskes (2015), "A multi-criteria Method for Automatic Web Page Summarization" NLP-RG, MIRACL Lab. University of Sfax, Tunisia.
3. Babar, S. A, and Thorat, S. A. (2017), " Improving text summarizationusing fuzzy logic & latent semantic analysis", *In international journal of innovative research in advance engineering (IJIRAE).* volume1 issue 4 (May 2017).
4. Suanmali, L., Salim, N., & Binwahlan, M. S. (2009), "Fuzzy logic based method for improving text summarization", arXiv preprint arXiv:0906.4690.
5. Ouyang Y, Li W, Li S, Lu Q (2011) , "Applying regression models to query-focused multi-document summarization", Inf Process Manag 47:227–237
6. Erkan, G., & Radev, D. R. (2004), "LexRank: Graph-based lexical centrality as salience in text summarization", Journal of Artificial Intelligence Research,22, 457-479.
7. Baralis E, Cagliero L, Mahoto N, Fiori A (2013) "GRAPHSUM : discovering correlations among multiple terms for graph-based summarization", Inf Sci 249:96–109 doi:10.1016/j.ins.2013.06.046
8. ZHANG Pei-ying, & LI Cun-he (2009), "Automatic text summarization based on sentences clustering and extraction", 978-1-4244-4520- 2/09/$25.00 @2009 IEEE.
9. Abbasi-ghalehtaki, R., Khotanlou, H., & Esmaeilpour, M. (2016), "Fuzzy evolutionary cellular learning automata model for text summarization", *Swarm and Evolutionary Computation.*
10. Erkan G, Radev D (2004), "LexRank: graph-based lexical centrality as salience in text summarization", J Artif Intell Res 22:457–479
11. Riedhammer K, Favre B, Hakkani-Tur D (2010), "Long story short-global unsupervised models for key phrase based meeting summarization", Speech Commun 52:801–815s
12. Suanmali, L., Salim, N., & Binwahlan, M. S. (2009), "Fuzzy logic based method for improving text summarization", arXiv preprint arXiv:0906.4690.
13. Fattah MA (2014), "A hybrid machine learning model for multi-document summarization", 592–600. doi:10. 1007/s10489-013-0490-0
14. Tsutomu HIRAO et al. (2014), "Extracting Important Sentences with Support Vector Machines", *NTT Communication Science Laboratories* 2-4, Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0237 Japan
15. Periantu Marhendri Sabuna et al. (2017), "Summarizing Indonesian Text Automatically By Using Sentence Scoring And Decision Tree", Universitas Atma Jaya Yogyakarta Yogyakarta, Indonesia
16. John M Conroy et al. (2001), "Text summarization via hidden Markov models", institute of advance computer studies university of Maryland collage park MD 20742 USA
17. Joel Larocca Neto et al. (2014), "Document Clustering and Text Summarization", Rua Imaculada Conceic¸ ˜ao 1155 Curitiba - PR, 80215-901. Brazil
18. Jiantao Hu et al. (2015), "A modified weighted TOPSIS to identify influential nodes in complex networks", School of Engineering, Vanderbilt University, TN 37235, USA
19. Amreen-Ahmad, & Tanvir-Ahmad, (2018), "A game theory approach for multi-document summarization", *Arabian Journal for Science and Engineering,* Springer

AUTHORS PROFILE

Faiyaz Ahmad, is working as an Assistant professror in department of Computer Engineering ,jamia Millia islamia, New Delhi

Yassar, is an M.Tech student at Jamia Millia Islamia in the discipline of Computer science & Engineering .His research domain include Data mining & Machine learning .

Amreen Ahmad is working as an assistant professor in department of Computer Engineering ,jamia Millia islamia