

Context-Aware Model of Abstractive Text Summarization for Research Articles

Gopinath Dineshnath^a, Selvaraj.Saraswathi^b

^aDepartment of Computer Science Engineering, Pondicherry Engineering College, Pillaichavady, Puducherry, 605014, India.

^bDepartment of Information Technology, Pondicherry Engineering College, Pillaichavady, Puducherry, 605014, India.

Abstract

Research article comprises of different sections each holds its own characteristic domain information. Summarization of entire article from multiple documents of multiple sections in precise form with special focus to contextual information is tedious. We proposed context-aware model to extract contextual texts from research article by utilizing multi-document directed graph for contextual matching phrases. We customized extractive summarization for abstractive text summarization with lucid-information as prime criterion. Decision matrix with elitism identification further fine tunes the abstractive text summary and outperforms at sentence level Rouge-L measures 9.32 and summary level measures 89.65.

Keywords

Glowworm Swarm Optimization, Analytical Hierarchical Processing (AHP), Skip Gram Model, Word2vectors, Maximum Marginal Relevance, Log-Likelihood Rate, N-Gram.

1. Introduction

Modern days, internet becomes the integral part of human and act as information highway. The primary source of information in digital world is Internet and it is boon for academicians, bloggers, students and researcher fraternity. Information available in Internet comprises of massive flow of information, which makes retrieval process complex with respect to context-specific content. Scientific article prevailing now with ocean of research domains makes difficult to scholar cope-up, grasp and streamline documents relevant to their interest. Query based search [1] for specific domain also fetch many relevant articles that is difficult task to categorize surpass human processing capabilities. In such scenario, automatic text summarization of articles is fruitful solution in terms of reducing time effort for reviewing entire articles and grab gist of information enclosed in it. Basically, summaries generation

in two ways; single-document summaries produce a summary from a given single source and multi-document summaries in which different but related documents are summarized by comprises only the vital materials or main ideas in a document in less space.

There is a vast difference between automatic multi-document summarization of generic texts to that of scientific articles. The major difference [2] between generic-text and research article summarization is; research article consists of different section namely abstract, introduction, literature survey, methods, results and discussions, whereas generic text's scope is extracted from first few sentences in first paragraphs and entire section holds at maximum 500 words.

In general, abstract and citation texts in scientific articles are considered for automatic summarization system.

Abstract section [3] is biased to author findings, author's own contribution, and evaluation metrics. In simpler way, abstract

ACI'21: Workshop on Advances in Computational Intelligence at ISIC 2021, February 25-27, 2021, Delhi, India.

Email: dinesh.gopinath60@gmail.com (G.Dineshnath); swathi@pec.edu (S.Saraswathi);

ORCID: 0000-0003-0026-1932



©2021 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

outlines the domain and list of findings in crisp manner depends upon the type of articles (review/original).

Citation sensitive or citation-based summaries^[4] is another type of Scientific article summarization, major task in summary production is clear-cut distinction of cited and non-cited text is performed. Citation summary system performs categorization of every sentence and labeled to citation or non-citation one. Later, evaluation measures based on similarity between each sentence in the reference article and citation sentences and then grouped it into one of two classes: cited or non-cited. Abstractive Multi document summarization [5] selectively picks up either first sentence of abstract or introduction of a paper since it is comprised of background information of research topic. Construction of appropriate title for article involves interpretation and integration of the concepts from multiple sentences of the abstract. Apart from that there exist multiple challenging issues like content organization, sentence compression and fusion, and paraphrasing sentences.

All summarization system should meet the summary length constraints as and other parameters specified by the user or summarization system is known as controllable summarization. Controllable summarization^[6] is the main criteria in summarization system which specifies the length of summary generation in accordance to the entities on which it focuses on and mimics the source's style. User may define the high-level attributes for summary generation. Summary generation is controlled by specific control variables, length, and source style, entities of interest and summarizing only remaining portions of the document. For instance, blog summarization, the primary thing is to derive representative words from comments and then selection of paramount sentences from the blog post which consists of representative words.

Context aware components are usually meant task of inferring contextual information. Contextual information detection might be detection ranging from topic community, paragraphs analysis, sentences and words by statistical computation measures. The most well-known computation is Set-Calculus techniques to find sentence similarity via Union, Jaccard Co-efficient, Cosine -Similarity measures followed by normalization

techniques. Our main focus is Context aware information inference from multiple documents.

2. Literature review

Lloret et al.^[7] have applied both extractive and abstractive summarization procedures for scientific article abstracts. The extractive summarizer (compendium E) is developed to perform conventional preprocessing such as breaking sentences, assigning tokens, stemming, lemmatization and PNG markers, tagging and removing duplicates at various sentence levels.^[8] A mixture of both extractive and abstractive technique (compendium E-A) is developed to support compendium E as base to incorporate sorted information which are relevant. Relevancy identification with respect to every sentences, assigns a score that emphasize its importance based on code quantity principle (CQP)^[9] compendium E-A derives abstractive summary by utilizing top ranked sentences with chronological ordering.

Saggion^[11] utilized pretrain models for learning and transformations for the problem of abstract generation. The initial summary generation from abstracts are generated and transformed to model based learning. The learning models assists with examples from corpus. Further, abstracts are gathered from GATE^[12] and Weka^[13] environment. Abstractive text summarization also known as natural language generation in natural language processing Paraphrasing of sentences is also another important criterion in natural language generation Paraphrasing of sentences^[14] involves substitution of relevant verbatim and modifying the direct to indirect speech or vice versa. The vector representation is purely focus on various sources of features namely LDA, D2V, W2V and encoding schemes^[15]. LDA^[16] explores semantic associations, D2V vectors finds contextual word vectors along with documents. Contextual aware model phase is concerned with contextual theme and dependency phrase extraction from multi documents using directed graph. minimal spanning tree is constructed for edges algorithm using Chu Liu Edmonds. (CLE)^[17] Knowledge base or ontology- based context aware model useful for domain classification and annotation.^[18] Evaluation of document summarization, Document

Understanding conference (DUC)^[19] benchmark datasets is used generally. Various datasets such as TAC^[20] shared task for text processing and document summarization. Similarly, DPIL^[21] for paraphrasing Indian languages for text summarization.

$$W(V_i) = (1 - d) + d * \frac{\sum_{V_j \in \text{In}(V_i)} \text{overlap}(s_i, s_j)}{\sum_{V_k \in \text{Out}(V_j)} \text{overlap}(s_j, s_k)} W(V_j) \quad (1)$$

The edge weights for nodes are assigned using status score on the basis of inwards and outwards edges, $W(V_i)$ represents status score assigned to vertex V_i . In (V_i) and Out (V_j) are inwards and outwards edges, points from particular node. After several iterations, each sentence in the document is assigned with a score. The top-n sentences are selected and ranked which constructs the summary for the document. There exist some dependent phrases in graph. The dependent phrases are cyclic in nature and some disjoint nodes. Such, disjoint nodes in graph are connected using CLE algorithm. On basis of lexical contexts, proposed procedure performs well than traditional keyword-based algorithms. We enhance the extractive summary production by adding co-occurrence measures to ensure Concept-based Point wise Mutual Information. (CPMI) CPMI weighs the different section in paragraphs gradually it weights decreases from beginning of paragraph to end of paragraph in document. CPMI measures support distributional semantics among phrases. CPMI weightage scheme $H(\beta)$ is expressed in equation 2.

$$H(\beta) = \begin{cases} C - \text{positive constant } \beta * B, & \text{if } \beta < -\left(\frac{\log B}{\log C}\right) \\ 1 & \text{otherwise} \end{cases} \quad (2)$$

3. Context Aware Model Phase

Context aware component model proposed follows contextual keyword interpretation, topic detection and topic clusters formation. Context aware component model also determines vector manipulation by bag of words and skip gram key terms with respect to specific documents. One hot encoding scheme

is used to slide over word vectors for prediction of context vectors. Skip gram models skips the selection of common words rather than contextual words. N gram models hold predefined size 'n' that triggers the selection of contextual words upon size limited to 'n'. Both skip gram and N gram models are desirable notion to pick keywords in context based on their lexicography collocations. The Proposed diagram is given in figure-1.

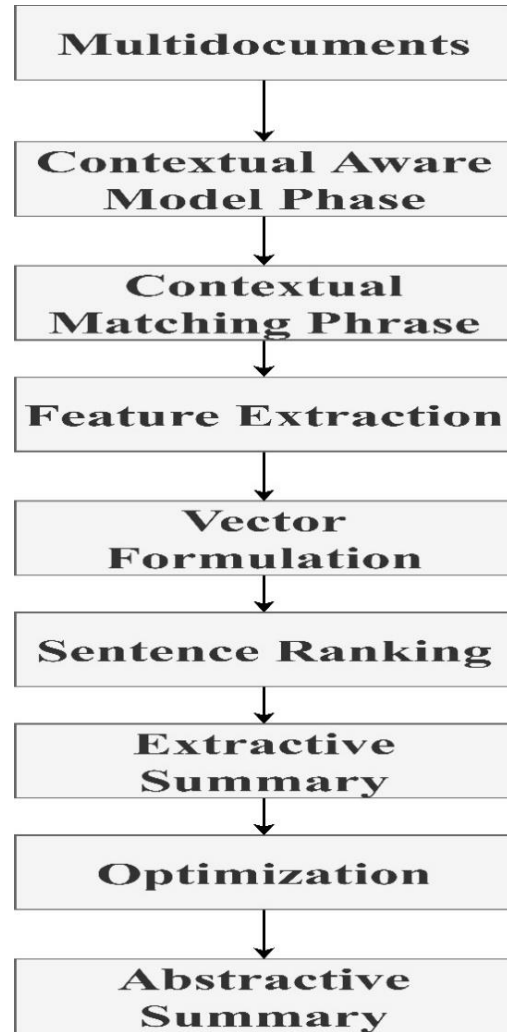


Figure 1: Proposed Context aware model for Abstractive Summarization

3.1. Contextual matching phrases

Contextual matching phrase from multi documents is proposed to retrieve thematic portions with similar sentence phrases sequentially from one document to another. The graph based contextual word is intuitive way to

Formula for Extraction of Features.	Eq n.o:
$T_i = \frac{\text{Number of title Features}}{\text{average length(Title)}}$	(3)
$\text{Tag sum} = \frac{1 - e^{-\alpha}}{1 + e^{\alpha}}$	(4)
$(S_{bi}) = \frac{\sum L(S_i)}{L_{\max}}$	(5)
$D_{sm}(S_i) = \frac{TW(S_i)}{TMax}$	(6)

represent sentences as node and corresponding contextual word as vertices. The path projecting from node to node via outgoing and incoming vertices provides a notion either matching target phrases or discriminating phrases. subsequent dependent phrases need to be included in directed graph. The procedure for contextual theme and dependency phrases extraction is shown below.

- Step 1 Accept text and store it in text buffer.
- Step 2 new word falls into below categories.
- Step 2 a): If first word then adds to graph G.
- Step 2 b): If fresh word then appends to G.
- Step 3 Go to step 2 until words overlap.
- Step 4 If overlap, status score using (1).
- Step 5 Extract similar texts and update in G.
- Step 6 Construct Digraph using CLE.
- Step 7 Do updates to infer adjacent edges.
- Step 8: Output the phrases.

3.2. Feature Extraction

The feature extraction for document summarization includes title feature, proper nouns, Named Entities Recognition (NER) and parts of speech tagging, sentence boundary analysis and distributional semantic analysis. Title feature scoring scheme is based on the ratio of mean number of titles present to that of average length of title. The formula for title feature is expressed in equation. (3). The proper nouns are generally recognized as title words and minimum number of words to accept as title. NER marks or labels the salient sentences which is considered for summary. The scoring scheme is expressed in equation. (4). where $\alpha = (t(s) - \mu) / \sigma$ (sigmoid function) aggregates mean count of regular expressions, case level and numeric literals. Sentence boundary calculation is expressed in equation (5). Distributional Semantic Analysis weighs thematic concepts to find word co-occurrence. It is scored using formula. (6). Column one represents features, similarly column two represents formulae used for computation and column three indicates optimal features

customized to produce extractive to abstractive summary.in Table-1. The optimal features and their convergence in Adaptive Glowworm Optimization (AGWO) is discussed in section 3.4.

Table 1 Optimal Features Extraction.

Features	Glowworm optimization
Title feature	Luciferin update phase
Named entity recognition and Tagging	Movement phase
Sentence boundary Distributional Semantic Analysis	Neighborhood Phase Words and conceptual level

3.3. Vector Formulation

Word embeddings are feature vectors that represent words holds the property that similar words have similar feature vectors. The question might rise in mind where the embeddings come from. The response is (again): they are learned from data. Several algorithms exist to learn word embeddings. We consider only one of them: word2vec, and sole version of word2vec called skip-gram, which is well-known and currently utilized in practice. Word embedding learning, our goal is to build a model which we can use to convert a one-hot encoding of a word into a word embedding. Let our dictionary contain 10,000 words or Giga-word Corpus. Skip-gram model performs for given sentence, selection of a word is feed into classifier, and predict words before and after the selected word in a fixed window. Negative sampling provides better vectors for frequent words with low dimension.

Latent Dirichlet (LD) allocation is a possibility-based mechanism viable for assortments, for example, text assortments. LD consolidates the documents as a blend of shifted topics; every unit involves words that have a spun affiliation that exists between them. Also, word choice simply dependent on the numerical idea of likelihood. Recursively determining the interaction of themes and words is done for the phase of a lonely record or a large number of

documents. At long last, yields the record which comprises of different subjects.

LD allocation algorithm performs the following:

1) Determine the number N of words in the document concerning probability distribution is Poisson.

2) Pick a merge of focuses for the report from a predefined set of K subjects as demonstrated by the Dirichlet movement.

3) Produce individually word in the list of terminology as follows language Vocabulary (V).

- a) Choose a subject;
- b) Choose a word in this subject.

3.4. Sentence Ranking

Sentence ranking phase, chiefly performs identification of prominent sentences with pertinent information and free from redundancy. It selects topmost sentences from documents and produces summary with application of traditional maximization algorithm like EM.^[23] finally produces extractive summary. However, Extractive summary lacks positional placement of sentences. hence there is a need to revisit sentence positions.^[24]

3.5. Extractive Summary

Extractive summary is created with top n sentences for research articles summarization Latter summary is transformed to decision matrix. Decision Matrix will keep track of extractive summary to make compatible for abstractive text summary with usage measure and penalty measure. Hence, optimization algorithm is used to remove exact replica of original text produced in extractive summary. Co-reference resolution is also handled in abstractive text summary generation. Meanwhile construction of cosine similarity was rapid, useful, and seems reasonable.

3.6. Optimization

$$li(t+1) = (1 - \rho)li(t) + \nu J(x_i(t+1)); \quad (7)$$

$$Ni(t) = \{j: dij < rid(t); li(t) < lj(t)\} \quad (8)$$

$$rid(t+1) = \min\{rs, \max\{0, rid(t) + \beta(\hat{n}t - |Np(t)|)\}\}; \quad (9)$$

Glowworm Optimization (GO)^[22] comprises of three phases namely Luciferin update phase, Neighborhood phase and movement phase. Adaptive Glowworm Optimization (AGO) is proposed for tailor-made features to acquire vectors or extract features to frame the summary. The optimization principle is based on five features and their application phase is listed in Table-1.

Sentence position is additional feature to revisit sentence with appropriate ordering. Positioning of sentences which is most vital part in the summary generation have higher weights. The feature associated with sentence length; hence we have minimal set of 25 words to accept as a sentence. F5 feature is in movement phase of Glow worm optimization with luciferin value or luminous quotient, affine towards the similar topics.

Luciferin update phase, sentences are concatenated with respect to the relevancy. Relevancy is determined by feature with respect to title and all sentences in document. Luciferin update phase, movement phase and neighbor phase are expressed as equation in (7) (8) and (9) respectively Luciferin enhancement(v) depends upon Proper Nouns.

J(xi(t))-objective function which maximize weights of every proper nouns. Luciferin decay constant gradually decreases when common noun exists.

Movement Phase, forms local clusters based on decision range. Sentences are of similar contexts likely to move based on entity-tagging features. Finally, neighbor Phase performs chronological sorting of clustered sentences to produces summary.

4. Results and Discussions

The extractive summarization shows better results with proposed procedure, the summary produced purely relies on lexical features and surpass traditional keyword ranking schemes. yellow color denotes dependency phrases and green color denotes contextual theme. The output of context aware component of extractive summarization is shown as well as output of abstractive text summarization is also shown.

4.1. Training and Testing

For training purpose, Document understanding Conference (DUC) data set taken into considerations. The precise explanation of DUC data sets and DUC data is customized, which is free from least significant words or stop words according to port-stemmer's algorithm. Recall oriented understudy Gist evaluation, Rouge(R) is also considered for evaluation. R falls into many variants like R-unigram, R-bigram, R-Longest common Subsequence and R-N gram classes.

Multiple documents of artificial intelligence domain for testing and performed various measures like R-1, R-2, and R-L (Longest Common Subsequence) scoring. At Sentence-level, computes longest common subsequence (LCS) between two pieces of text ignores new lines and summary-level, newlines in the text are interpreted as sentence boundaries, and the LCS is computed between each pair of reference and candidate sentences, and their results are tabulated below in table-2.

The proposed AGO performs well in sentence and summary level than traditional methods. similarly, contextual theme detection also out performs than traditional schemes like lexrank, maximum relevance, loglikelihood ranking [25],[26],[27] and other centrality measures as baseline evaluation.

Table 2 Results Comparison

Methods	R-1	R-2	R-L Sentence level	R-L Summary level
Proposed AGO	0.4611	0.1342	0.932	0.89
MMR	0.3716	0.0757	0.16	0.80
LEX	0.4030	0.0913	0.69	0.53
Proposed contextual theme	0.4297	0.0835	0.12	0.83
Tf-idf	0.3639	0.0736	0.14	0.81
LLR	0.3975	0.0850	0.084	0.64

Output of Context aware Extractive and Abstractive Summarization

Extractive Summary

A large number of methods applied in the field of extractive summarization over the years. Scoring sentences for such summary is tedious task. Many researchers putting so much effort to improve the quality of summary. Document summarization focus both quality and coverage of content. Clustering of sentences in document summarization shown promising results to discover topics. A Fuzzy oriented clustering for summarization of multi-documents. Compendium- a summarizer tool, generates relevant summary free from redundancy Collabsum clustering process both inter and intra document relationship and forms clusters. Clusters in turn apply graph based ranking methodology. FEOM...genetic algorithm...graph -based approach...probabilistic model.

Abstractive summary

Various types of Sentence clustering techniques applied to document summarization Sentence scoring, topic coverage, relevant sentences and summarization quality are main components in summary production. Clustering algorithms for sentence scoring and grouping similar sentences according to topics conveyed in document. A fuzzy based, evolutionary based clustering also successfully applied in conjunction with other graph-based approaches to provide summary.

5. Conclusion

Abstractive text summarization for research articles generates sentences individually using glowworm optimization with six associated features. In addition, decision matrix with elitism identification is formulated to choose summary sentences from both extractive summary sentences and abstractive summary sentences with consistency as necessary condition. Extractive summary is reduced to more than 80% to generate abstractive summary. Extractive summary with 661 word tokens is produced as output in first phase. Later, decision matrix with Elitism identification produces abstractive summary with 84 tokens is obtained as final output. Proposed multi-document directed graph contextual matching phrases, Rouge-L measures in sentence level is 12.08 and Rouge-L measures in summary level is 83.95 for extractive summary. Similarly, Rouge-L measures in sentence level is 9.32 and Rouge-L measures in summary level is 89.68 for abstractive summary. A novel model has been implemented to be ample enough to provide multi objectives and to convince the instantaneous needs. Ultimately, this study will inspire many researchers to further explore and apply the various types of Swarm intelligence while solving the summarization tasks, specifically in the abstractive text summarization (ATS) field.

6. Future works

Decision matrix performs combination of Sentences from extractive summary are assessed and deemed to be fit for abstractive summary are analyzed in conjunction with input from optimization algorithm with associated six features. Selection of best sentences and worst sentences based on their usage and penalty is awarded to compose summary. Global decision matrix performs elitism identification (algorithm) and outputs sentences with sentence flow as criterion. However, decision matrix follows Analytical Hierarchical Processing (AHP) ^[28] with user defined decision values and their decisions are normalized. We can extend the normalized vectors by using fuzzy ^[29] based membership assessment as stated by Charugupta,et.al.^[30]

References

- [1] Shafiei Bavani, Elaheh, Mohammad Ebrahimi, Raymond Wong, and Fang Chen. "A query-based summarization service from multiple news sources." In 2016 IEEE International Conference on Services Computing (SCC), IEEE, (2016): pp.42-49. doi: 10.1109/SCC.2016.13.
- [2] Bharti, Santosh Kumar, Korra Sathya Babu, Anima Pradhan, S. Devi, T. E. Priya, E. Orhorhoro, O. Orhorhoro, V. Atumah, E. Baruah, and P. Konwar. "Automatic keyword extraction for text summarization in multi-document e-newspapers articles." *European Journal of Advances in Engineering and Technology* (2017): 4, pp.410-427.
- [3] Chowdhury, S. M., and Mazharul Hoque. "A Review Paper on Comparison of Different Algorithm Used in Text Summarization." *Intelligent Data Communication Technologies and Internet of Things: ICICI 2019 38* (2019): 114.
- [4] Cohan, Arman, and Nazli Goharian. "Scientific article summarization using citation-context and article's discourse structure." (2017). arXiv preprint arXiv:1704.06619
- [5] Nouf Ibrahim Altmami, Mohamed El Bachir Menai, "Automatic summarization of scientific articles: A survey." *Journal of King Saud University-Computer and Information Sciences*, (2020) doi.:10.1016/j.jksuci.2020.04.020.
- [6] Fan, Angela, David Grangier, and Michael Auli. "Controllable abstractive summarization." arXiv preprint arXiv:1711.05217 (2017).
- [7] Lloret, Elena, María Teresa Romá-Ferri, and Manuel Palomar. "COMPENDIUM: A text summarization system for generating abstracts of research papers." *Data & Knowledge*

- Engineering (2013):88, pp.164-175.
- [8] Ferrández, Oscar, Daniel Micol, Rafael Munoz, and Manuel Palomar. "A perspective-based approach for solving textual entailment recognition." In Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing, (2007). pp.66-71.
- [9] Gardenfors, Peter. *The geometry of meaning: Semantics based on conceptual spaces*. MIT press, 2014.
- [10] Luhn, Hans Peter. "The automatic creation of literature abstracts." *IBM Journal of research and development* (1958): 2, pp.159-165.
- [11] Saggion, Horacio. "Learning predicate insertion rules for document abstracting." In *International Conference on Intelligent Text Processing and Computational Linguistics*, Springer (2011) pp.301-312.
- [12] Maynard, Diana, Valentin Tablan, Hamish Cunningham, Cristian Ursu, Horacio Saggion, Kalina Bontcheva, and Yorick Wilks. "Architectural elements of language engineering robustness." *Natural Language Engineering* (2002): 8, pp.257-274.
- [13] Witten, Ian H., and Eibe Frank. "Data mining: practical machine learning tools and techniques with Java implementations." *ACM Sigmod Record* (2002):31, pp.76-77.
- [14] Sethi, Nandini, Prateek Agrawal, Vishu Madaan, and Sanjay Kumar Singh. "A novel approach to paraphrase Hindi sentences using natural language processing" *Indian Journal of Science and Technology* (2016):9(28), pp.1-6.
- [15] Alguliyev, Rasim M., Ramiz M. Aliguliyev, Nijat R. Isazade, Asad Abdi, and Norisma Idris. "COSUM: Text summarization based on clustering and optimization." *Expert Systems* (2019):36, doi:10.1111/exsy.12340.
- [16] Gupta, Monika, and Parul Gupta. "Research and implementation of event extraction from twitter using LDA and scoring function." *International Journal of Information Technology* (2019):11, pp.365-371.
- [17] Nizami, Muhammad, and Ayu Purwarianti. "Modification of Chu-Liu/Edmonds algorithm and MIRA learning algorithm for dependency parser on Indonesian language." In *2017 International Conference on Advanced Informatics, Concepts, Theory, and Applications (ICAICTA)*, IEEE (2017) pp.1-6.
- [18] Malik, Sonika, and Sarika Jain. "Ontology based context aware model." In *2017 International Conference on Computational Intelligence in Data Science (ICCIDS)*, IEEE (2017). pp.1-6.
- [19] Sanchez-Gomez, Jesus M., Miguel A. Vega-Rodríguez, and Carlos J. Pérez. "Extractive multi-document text summarization using a multi-objective artificial bee colony optimization approach." *Knowledge-Based Systems* (2018): 159, pp.1-8.
- [20] ShafieiBavani, Elaheh, Mohammad Ebrahimi, Raymond Wong, and Fang Chen. "A graph-theoretic summary evaluation for rouge." In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (2018): pp.762-767.
- [21] Anand Kumar M., Singh S., Kavirajan B. and Soman K.P. "Shared Task on Detecting Paraphrases in Indian Languages (DPIL): An Overview". In: Majumder P., Mitra M., Mehta P., Sankhavara J. (eds) *Text Processing. FIRE 2016. Lecture Notes in Computer Science*, Springer (2018):10478.
- [22] Alphonsa, MM Annie, and P. Amudhavalli. "Genetically modified glowworm swarm optimization based privacy preservation in cloud computing for healthcare sector." *Evolutionary Intelligence* (2018):11 pp: 101-116.
- [23] Janani, R., and S. Vijayarani. "Text document clustering using spectral clustering algorithm with particle swarm optimization." *Expert Systems with Applications, Elsevier* (2019) 134, pp.192-200.
- [24] Xu, Song, Haoran Li, Peng Yuan, Youzheng Wu, Xiaodong He, and Bowen Zhou. "Self-Attention Guided Copy Mechanism for Abstractive Summarization." In *Proceedings of the*

- 58th Annual Meeting of the Association for Computational Linguistics (2020): pp.1355-1362.
- [25] Weng, Shi-Yan, Tien-Hong Lo, and Berlin Chen. "An Effective Contextual Language Modeling Framework for Speech Summarization with Augmented Features." In 2020 28th European Signal Processing Conference (EUSIPCO), IEEE (2021): pp.316-320.
- [26] Mallick, Chirantana, Ajit Kumar Das, Madhurima Dutta, Asit Kumar Das, and Apurba Sarkar. "Graph-based text summarization using modified TextRank." In *Soft computing in data analytics*, Springer (2019): pp.137-146.
- [27] Sabbah, Thabit, Ali Selamat, Md Hafiz Selamat, Fawaz S. Al-Anzi, Enrique Herrera Viedma, Ondrej Krejcar, and Hamido Fujita. "Modified frequency-based term weighting schemes for text classification." *Applied Soft Computing* (2017):206, pp.58-193.
- [28] Tofighy, Seyyed Mohsen, Ram Gopal Raj, and Hamid Haj Seyyed Javad. "AHP techniques for Persian text summarization." *Malaysian Journal of Computer Science* (2013): 26, pp. 1-8.
- [29] Bansal, Neha, Arun Sharma, and R. K. Singh. "Fuzzy AHP approach for legal judgement summarization." *Journal of Management Analytics* (2019):6, pp.323-340.
- [30] Gupta, Charu, Amita Jain, and Nisheeth Joshi. "Fuzzy logic in natural language processing—a closer view." *Procedia computer science* (2018):132 pp.1375-1384.