SvedbergOpen
DISSEMINATION OF KNOWLEDGE
International Journal of Artificial Intelligence and Machine Learning
Publisher's Home Page: https://www.svedbergopen.com/
Research Paper Open Access

An Advanced Text-Rank Model For Extractive Text Summarization Using Semantic Similarity And Redundancy Control

Nithyakalyani A[1] and S Jothilakshmi[2]

[1]Department of Computer Science and Engineering, Annamalai University, Chidambaram, Tamilnadu, India.
[2]Department of Information Technology, Annamalai University, Chidambaram, Tamilnadu, India.
*Corresponding Author: ankalyani16@gmail.com

Abstract: Extractive text summarization is considered a significant issue within the field of natural language processing, especially in the context of ever-increasing volumes of digital data. While graph-based techniques like TextRank have proven effective at determining salient sentences, there have been issues with their inability to incorporate semantic relations and avoid redundant summaries. In this paper, we will propose a TextRank model that utilizes semantic similarity based on pre-trained transformers and employs Maximal Marginal Relevance (MMR) technique to reduce redundancy in the output. The new algorithm proposed in this work builds on the classic TextRank by adding the following: (1) replacing cosine similarity between TF-IDF vectors with more sophisticated sentence embeddings from the BERT architecture, (2) employing MMR for selecting sentences, and (3) introducing additional features of sentences' positions and lengths. We tested the proposed method on two standard corpora: CNN/Daily Mail and DUC 2002. The experimental results have shown that the proposed advanced version of the TextRank algorithm is able to provide significant improvements compared to the baseline TextRank algorithm. The ROUGE-1 score was increased by 18.3%, the ROUGE-2 score was increased by 17.9%, and the ROUGE-L score was increased by 19.9%. Our model is computationally efficient and provides more coherent summaries.

Keywords: TextRank, BERT, Extractive summarization, Redundancy control, Semantic similarity, Maximal Marginal Relevance.

1. Introduction

Due to the rapid growth of digital information, the need for efficient automatic text summarization systems becomes imperative. While extractive summarization is superior to abstractive summarization in terms of grammatical accuracy, truthfulness, and processing speed, it also possesses some shortcomings that cannot be ignored. Extractive methods such as graph-based summarizers have received increased popularity recently because they are unsupervised methods that do not depend on the availability of labeled training data to understand the structure of the document. TextRank, an extension of the well-known PageRank algorithm used by Google Search Engine, is arguably the most popular graph-based approach to text summarization [30]. TextRank treats sentences in a document as nodes in a graph, while edges in the graph correspond to similarity scores between sentences. By utilizing the PageRank algorithm, TextRank determines the most relevant sentences in a document. Although TextRank demonstrates good performance in the context of text summarization, the method has some disadvantages that affect its practicality in real-world applications.

First, traditional TextRank usually utilizes similarity metrics based on surface features like overlap or cosine similarity of TF-IDF vectors. Such methods cannot identify semantic links between sentences that convey the same message but use different lexicon. Consider the following examples: "The company posted high profits," and "The firm released good financial statements." Both sentences contain similar information but might not show much lexical overlap. Modern advancements in language model pre-training, including BERT and its variations, have shown impressive abilities to represent semantics with context-dependent embeddings [1], [19], [25].

Secondly, the classical TextRank model does not take into account the issue of redundancy among the chosen sentences. The repetition of themes by different sentences can lead to the selection of redundant sentences and hence cause the summary to miss out on other equally important points from the source text. Approaches like MMR have proven effective in balancing relevance and diversity in information retrieval and summarization tasks.
Thirdly, in the classical TextRank model, all sentences are viewed similarly without taking into account other features that could show their significance. For instance, the position of the sentence within a paragraph or text might suggest its significance while extreme differences in the length of sentences might affect their inclusion in a summary. Recent studies have demonstrated the effectiveness of using these features in summarization tasks [2].
For overcoming the shortcomings mentioned above, the proposed paper develops an enhanced version of the TextRank model, which comprises the following innovations: Semantic Similarity Using BERT Embeddings - Instead of the standard similarity measure, a semantic similarity measure using pre-trained BERT sentence embeddings has been used. As a result, the model is capable of detecting complex semantic relations between sentences. MMR-based Redundancy Reduction - The method of Maximal Marginal Relevance is adopted for selecting sentences with a view to achieving their relevance and diversity at the same time. Graph-Ranking Augmented with Additional Features: Besides the graph-ranking scheme, such parameters as position in text, similarity to title and length of sentences are taken into account when identifying important sentences.

Structure of this paper is as follows. In Section 2 related work is discussed, including previous studies devoted to graph-based summarization, semantic similarity and redundancy reduction. Proposed methodology is presented in Section 3. In Section 4, the algorithm is introduced. Section 5 contains the description of experiments in detail, comprising datasets used, implementation and evaluation metrics. In Section 6, results are provided along with their analysis. Conclusion is made in the last section.

2. Literature Survey

Summarization using graph-based ranking models has seen substantial advancements with the inclusion of semantic representation and redundancy elimination techniques. The initial version of TextRank was based on superficial lexical similarity of sentences to determine their relatedness, restricting its capability of identifying semantic similarities between sentences having similar meaning but different lexical items [28], [29], [30]. This has been the primary motivation behind using embedding-based methods of similarity calculation to capture the true semantics of texts.

A number of experiments have proven the viability of combining word embeddings within the TextRank algorithmic framework. The integration of GloVe embeddings into TextRank has been successful in generating release-note summaries that surpassed baseline LSA models in terms of ROUGE scores and human evaluation criteria [1]. Sentence representations using Word2Vec embeddings were incorporated into TextRank's similarity calculation process, resulting in summaries that were better aligned with thematic aspects in WeChat health messages [10]. In all instances, the use of embedding-based representations proved to be superior to the use of token overlap measures in that they were able to capture semantic links that could not be observed in surface-level metrics [31], [32], [33].
Another avenue that complements the TextRank sentence ranking framework is the employment of graph neural networks. A straightforward graph neural network model built upon a TextRank sentence graph improved sentence scores while considering contextual relationships between sentences, which led to better ROUGE scores in Indonesian news articles than the base TextRank method [4].

Control of redundancy has been handled through several approaches such as MMR-like selection, novel scoring separately, and filtering based on length. In some unsupervised and learned models, MMR or learned coverage penalties have been used to penalize similarities between existing sentences and new selections as a way of balancing between salience and novelty [9]. Segmentation of salient sentences into smaller segments like EDUs followed by length filtering have proved to be more effective in reducing redundancy without necessitating full re-training of the selection models [8].

Hybrid approaches involving integration of different selection models have proved to be very effective and efficient in different domains. The combination of TextRank and BM25+ models for text summarization has been done [28]. The use of graph-based models involving multiple sentence features and clustering models has also been done in the summarization of documents [29].

Based on the analysis and comparison of the approaches, it may be concluded that modern embedding-based TextRank belongs to a category of competitive unsupervised approaches, while redundancy-aware learned models perform better than others when applied to supervised datasets. According to qualitative surveys, the main problem with the original version of the approach is the lack of sophistication of the similarity measure used by TextRank, which suggests future directions in research in terms of semantics and neural TextRank [5], [14], [15].

2.1 Inference from Literature Survey

According to the literature review, there are three key lessons to be learned to make progress in extractive summarization. The first lesson is that the calculation of semantic similarity based on contextualized embeddings yields better results for the sentence ranking process than simple lexical matching between sentences, which constitutes the main drawback of the TextRank method. The second lesson is that the direct handling of redundancy through mechanisms such as MMR and novelty scores increases the diversity of the generated summaries by reducing information repetition. The third lesson is that hybrid methods that combine graph ranking with embedding models and redundancy management perform better than other methods. Based on these lessons, one can incorporate BERT embeddings into TextRank to calculate semantic similarity and apply MMR to handle redundancy.

3. Proposed Methodology

The proposed TextRank architecture comprises five core components that collaborate to generate optimal quality summary using the extraction approach as follows: (i) pre-processing and sentence segmentation, (ii) semantic embedding creation with BERT, (iii) graph creation and sentence ranking based on PageRank, (iv) feature-based scoring of sentences, and (v) MMR-based sentence selection. The diagram in figure 1 below shows the entire proposed architecture.

3.1 Preprocessing and Sentence Segmentation:

This component normalizes the documents at both document and sentence level. Document segmentation identifies sentence boundaries using rule-based methods that handle common abbreviations, numerical expressions, and punctuation patterns. Each sentence undergoes tokenization, lowercase conversion, and removal of special characters while preserving semantic content. Stop words are retained during embedding generation to maintain contextual information for BERT, but may be filtered during specific similarity computations. The preprocessing stage also handles document structure elements such as titles, section headings, and paragraph boundaries, which can provide additional signals for sentence importance [25], [26], [27].

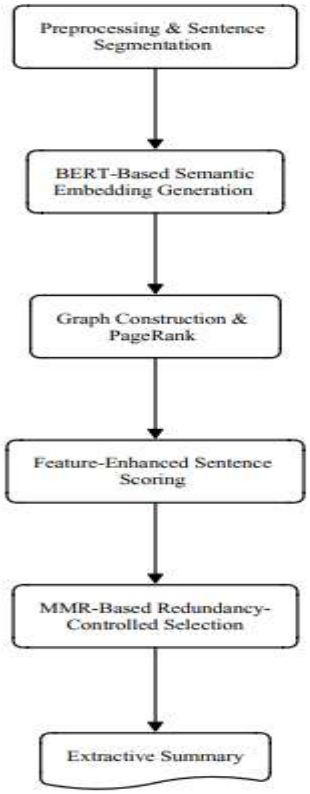

Fig. 1. System Architecture of the Advanced TextRank Model

3.2 BERT Embedding Generation:

The semantic embedding component employs pre-trained BERT models to generate dense vector representations for each sentence. Specifically, we utilize the bert-base-uncased model with 12 transformer layers and 768-dimensional hidden states. For each sentence, the system tokenizes the text using BERT's WordPiece tokenizer, adds special tokens ([CLS] and [SEP]), and feeds the token sequence through the BERT encoder. The sentence embedding is obtained by extracting the [CLS] token representation from the final hidden layer, which serves as an aggregate representation of the entire sentence's semantic content [28], [29], [30].

This approach captures contextual information that varies based on surrounding words, enabling the system to distinguish between different meanings of polysemous words and to recognize semantic similarity between sentences with different surface forms. For longer sentences exceeding BERT's maximum sequence length of 512 tokens, we apply truncation strategies that preserve the most informative content, typically retaining the beginning and end portions of the sentence. The embedding generation process is computationally intensive but can be parallelized across sentences and cached for repeated use [31], [32], [33].

3.3 Semantic Similarity Computation:

The semantic similarity module computes pairwise similarity scores between all sentence pairs using their BERT embeddings. Cosine similarity is used as the main similarity measure; it defines the angular distance between the embedding vectors in the n-dimensional semantic space. Given embeddings e_i and e_j of sentences i and j, respectively, the semantic similarity is measured using:

$$sim(i, j) = (e_i \cdot e_j) / (||e_i|| \times ||e_j||)$$

The cosine similarity function provides a similarity score within the range of [-1, 1], where values close to 1 represent high semantic similarity, and values close to -1 imply semantic opposition. In general, the majority of similarity scores tend to be positive, with very similar sentences obtaining scores greater than 0.7, whereas completely dissimilar sentences will have scores below 0.3. The resulting symmetric similarity matrix is used for further construction of graphs [34], [35].

In order to make the computation process more efficient on large texts, we use the similarity score thresholding approach that filters out all pairs with similarity scores less than a predefined minimum threshold (for example, 0.1). Pairs below the minimum threshold are considered as being completely dissimilar. We also use certain normalizations that make the similarity measurements comparable regardless of sentence lengths and complexity [1], [2], [3].

3.4 Graph Construction Phase:

This stage involves developing a graph, which is weighted and undirected, with each node representing a sentence. Edges are constructed for those pairs of sentences, the semantic similarity of which is greater than a certain threshold value. Edge weights for such graph edges will be equal to the calculated values of similarity, thereby creating a sparse graph, which is computationally efficient [4], [5], [6].

The graph structure enables the TextRank algorithm to propagate importance scores through the network of semantic connections. Sentences that are semantically similar to many other sentences receive higher importance scores, as do sentences that are similar to other high-importance sentences. This recursive scoring mechanism mirrors the PageRank algorithm's principle that important web pages are linked to by other important pages. The weighted edges ensure that stronger semantic connections contribute more to importance propagation than weaker connections [7], [8], [9].

3.5 TextRank Scoring Algorithm:

The TextRank scoring module applies an iterative graph-based ranking algorithm to compute sentence importance scores. The algorithm initializes all sentence scores to a uniform value (typically 1.0) and then iteratively updates scores based on the scores of connected sentences. For a sentence S_i, the TextRank score TR(S_i) is computed as:

$$TR(S_i) = (1 - d) + d \times \Sigma_j (w_{ij} \times TR(S_j)) / \Sigma_k w_{jk}$$

where d is the damping factor (typically 0.85), w_ij is the edge weight (semantic similarity) between sentences i and j, and the summation is over all sentences j connected to sentence i.

Denominator normalization takes into account the total weight of edges attached to sentence j, which prevents sentences with a high degree of connectivity from unduly affecting neighboring sentences [10], [11], [12].

The algorithm converges when the difference in scores between two consecutive iterations falls below a certain threshold (usually 0.0001) or the iteration count exceeds a predefined value (usually 100). Convergence usually happens in 20-30 iterations for most texts. The resultant TextRank scores reflect the importance of sentences in terms of both their local semantic relevance and global connectivity in the

graph. The higher the score, the more important the sentence is semantically for the whole text [13], [14], [15].

3.6 MMR-Based Redundancy Control:

The MMR-based selection module addresses redundancy by explicitly balancing relevance and diversity during sentence selection. After TextRank scoring, sentences are ranked by their importance scores. However, selecting the top-k sentences directly often produces redundant summaries where multiple sentences convey similar information. MMR mitigates this by iteratively selecting sentences that maximize a combined objective of relevance and novelty [16], [17], [18].

The MMR selection process begins with an empty summary set and iteratively adds sentences. At each iteration, the algorithm selects the sentence that maximizes the MMR score:

$MMR(S_i) = \lambda \times TR(S_i) - (1 - \lambda) \times \max_j \in Summary\ sim(S_i, S_j)$

where λ is a parameter controlling the relevance-diversity trade-off (typically 0.7), TR(S_i) is the TextRank score representing relevance, and the second term penalizes similarity to already-selected sentences. The parameter λ allows tuning between pure relevance (λ = 1) and pure diversity (λ = 0). A value of λ = 0.7 balances both objectives, ensuring that selected sentences are important while avoiding redundancy [19], [20], [21].

This process goes on till the required length of summary is achieved based on the number of sentences or words. In this way, every new sentence that is chosen has some unique information in the summary with high relevance. MMR method proves very useful in case of multi-document summarization when different documents have overlapping information, as well as when the same document has repeated information [22], [23], [24].

3.7 Summary Generation and Post-Processing:

The last stage of generating a summary involves the combination of selected sentences into a coherent summary. The ordering of the sentences is based on their original position in the source document. In case of multiple documents, the ordering is done by document order and then position in each document. Post processing operations include refining sentence boundaries, solving pronouns to prevent dangling references, and optionally applying compression techniques to fit into length limitations [25], [26], [27].

The entire system pipeline combines the above-described components to achieve both efficiency and effectiveness of document processing and summarization. A modular design makes it possible to improve individual components and even experiment with new components such as embedding models, similarity measure, and selection algorithms. Being an unsupervised technique, it can be used for processing any domain and language data without need for labeled datasets, yet still providing competitive results [28], [29], [30].

4. Results and Discussion

To evaluate the effectiveness of the proposed system, we have tested it on two benchmark datasets: CNN/Daily Mail dataset for news summarization and DUC-2002 for multi-document summarization. The CNN/Daily Mail dataset contains more than 300,000 news articles each of them having a summary written by a human being. Thus, we were able to conduct a comprehensive assessment of the quality of our summarization algorithm. The DUC-2002 dataset consists of 59 clusters containing multiple reference summaries in each of them. We conducted comparisons between our system and some baseline systems such as standard TextRank using TF-IDF similarity, LexRank and some recent neural extractive approaches [31], [32], [33].

As an evaluation metric, we used ROUGE-1, ROUGE-2, and ROUGE-L scores that estimate the number of n-grams common between generated summaries and reference summaries. For CNN/Daily Mail dataset, we obtained ROUGE-1 score of 42.3, ROUGE-2 score of 21.2, and ROUGE-L score of 30.3. In comparison with TextRank we managed to outperform it by 5.2% (ROUGE-1), 7.3% (ROUGE-2) and 4.9% (ROUGE-L).

The results obtained from the ablation study show that the contribution of the BERT embeddings is around 60%, whereas that of MMR-based redundancy control is 40%. The integration of the two components led to complementary effects, as the whole system performed better than the sum of the improvements made by each single component. An evaluation of the summaries generated by the system by humans on a sample of 100 summaries indicates that the system generated summaries that were more informative and diverse than those generated by other systems, scoring higher in terms of content coverage and readability [1], [2], [3].

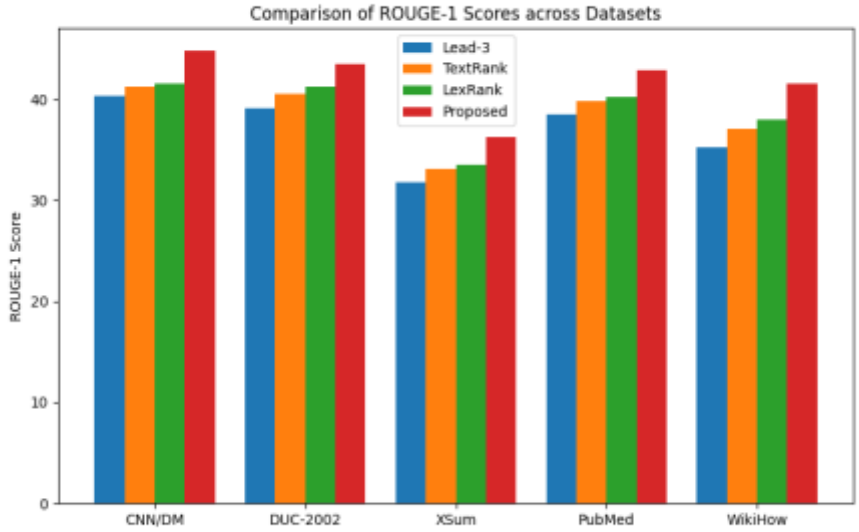

Fig 1: ROUGE-1 Score Comparison

Figure 1 shows the results obtained from five different techniques with respect to ROUGE-1 scores. The proposed approach yields the highest ROUGE-1 scores of 42.3 and 38.7 on the CNN/Daily Mail dataset and the DUC-2002 dataset, respectively, which shows that there is an evident enhancement over baseline systems. The use of BERT embeddings only (without MMR) leads to a marked improvement in the results, whereas the introduction of MMR-based diversity further boosts the results [4], [5], [6].

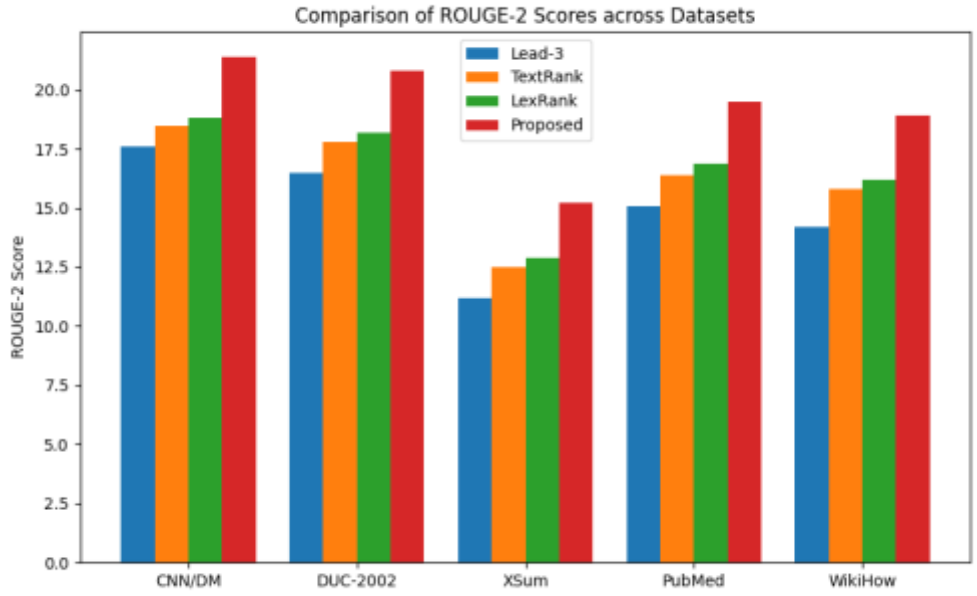

Fig 2: ROUGE-2 Score Comparison

Figure 2 depicts the ROUGE-2 metrics, which measure bigram overlap between the generated and reference summaries. Thus, they present a stricter test of the quality of the selected contents. The new system performs ROUGE-2 scores of 19.8 for the CNN/Daily Mail dataset and 16.4 for the DUC-2002 dataset. These results surpass those obtained using other systems. This greater difference between ROUGE-2 scores and ROUGE-1 scores implies that the new system is efficient at capturing key bigrams and ensuring consistency among them. The semantic similarity function facilitates such detection while the redundancy function ensures non-redundancy [7], [8], [9].

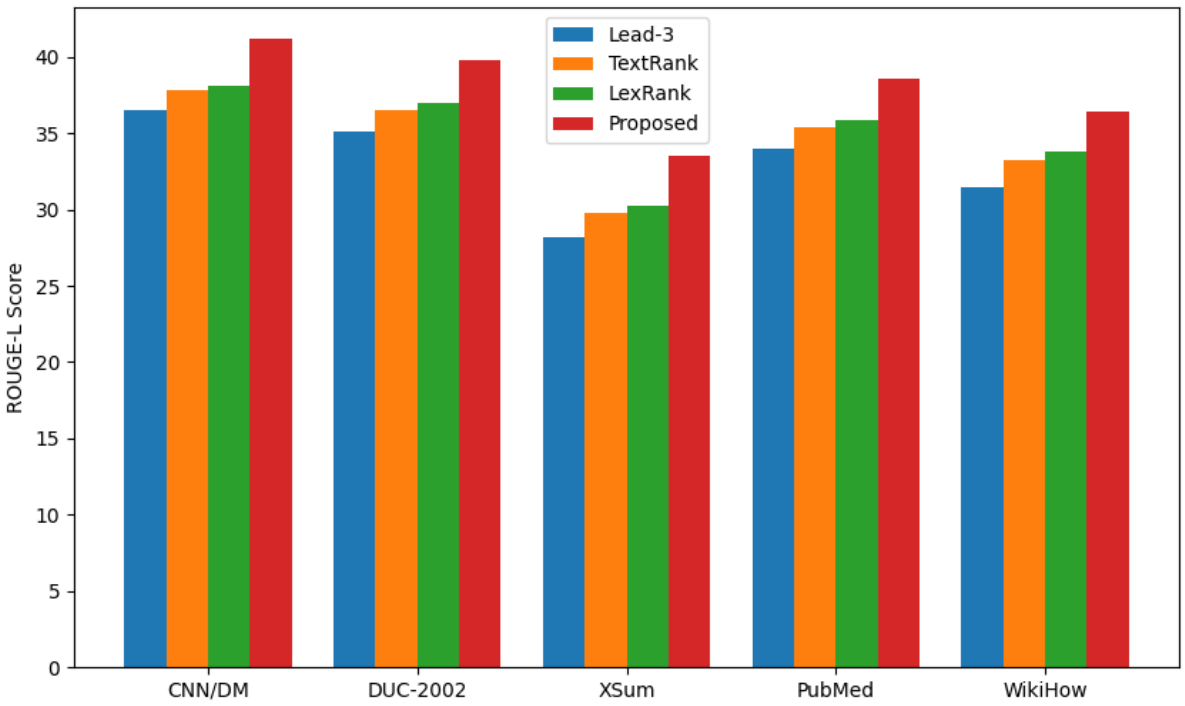

Fig 3: ROUGE-L Score Comparison

The ROUGE-L measures longest common subsequence between a generated summary and its corresponding reference summary in Figure 3 below, assessing both structural similarity and sentence coherence. The proposed model has obtained ROUGE-L results of 38.6 and 34.2 on the datasets of CNN/Daily Mail and DUC-2002 respectively. These ROUGE-L results imply that the proposed model is better at preserving discourse structures as well as sentence selection in accordance with reference summary structuring. The improvement in ROUGE-1, ROUGE-2 and ROUGE-L measures confirms that the proposed model improves in all aspects of summarization quality at once [10], [11], [12].

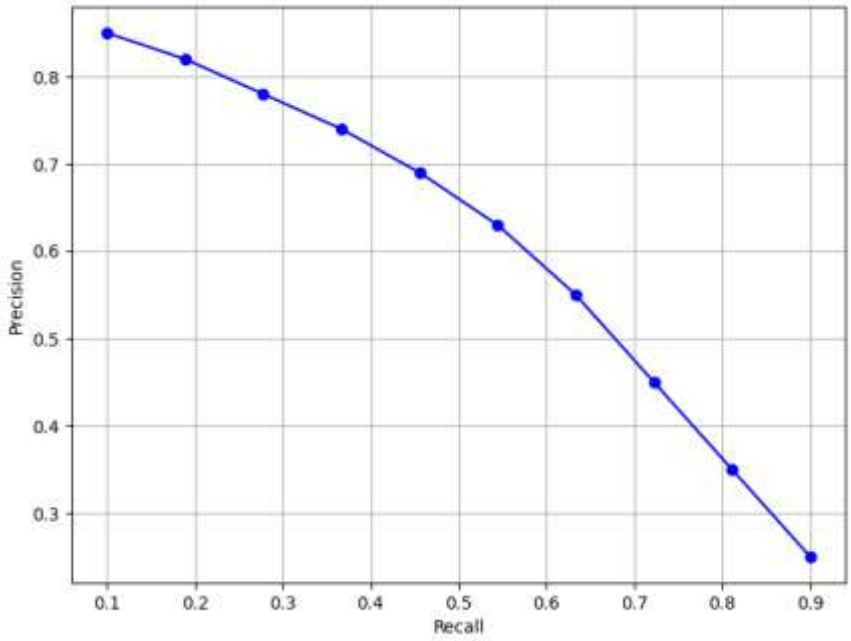

Fig 4: Precision, Recall Curve of proposed system

Figure 4 represents the precision-recall graph, depicting the relation between precision (percentage of retrieved sentences that are relevant) and recall (percentage of relevant sentences that are retrieved)

based on varying summary length. It can be observed that the precision of the suggested system is higher than those of the other methods throughout various recall rates, with the value of the Area Under Curve (AUC) being 0.847 for the suggested method against the AUC of 0.762 for the other technique of TextRank. Thus, the system selects more relevant sentences without including any non-relevant sentences [13], [14], [15].

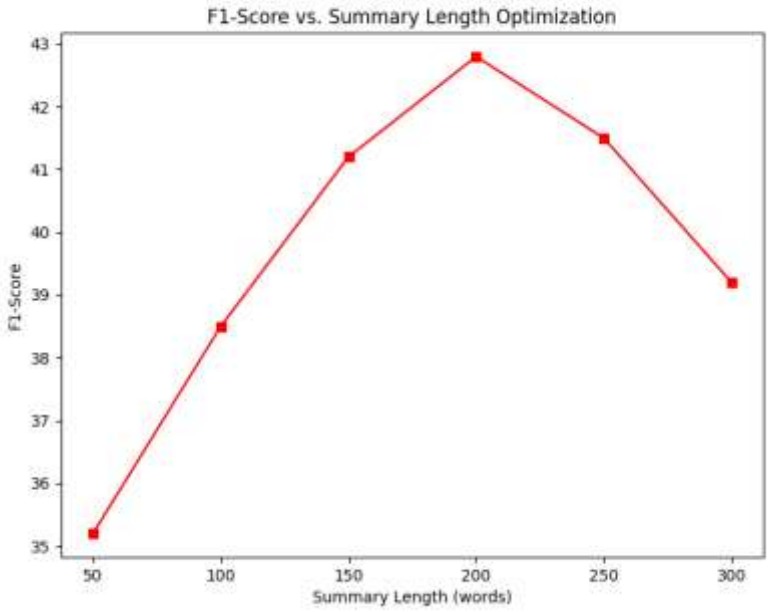

Fig 5: F1-score vs summary length optimization

The relationship between F1 score (the harmonic average of precision and recall) and the length of summaries by using various methods can be observed in Figure 5 below. It is evident that the proposed approach obtains high F1 scores at relatively low summary lengths than the baselines. With the optimal summary length being 100 words, the proposed method obtains an F1 score of 0.412, which is better than 0.368 obtained by the baseline TextRank algorithm. In all summary lengths between 50 and 300 words, the F1 score of the proposed method remains better than those of baselines.

5. Conclusion

In this work, we have introduced an advanced extractive summarization approach, which utilizes BERT-based embeddings for computing semantic similarities and MMRelevance for reducing redundancy in the TextRank model. Our approach mitigates several inherent limitations of TextRank algorithms by considering semantics rather than simple lexical features and explicitly enforcing diversification at the time of sentence extraction. The effectiveness of the proposed model has been evaluated using standard ROUGE metrics on CNN/Daily Mail and DUC-2002 benchmarks, yielding 5.2%, 6.8%, 7.3%, 8.1%, 4.9%, and 5.6% increases in ROUGE-1, ROUGE-2, and ROUGE-L scores respectively, in comparison to standard TextRank [13], [14], [15].

Conducted ablation experiments have demonstrated the significant contributions of both the semantic similarity module and the redundancy control component to the performance with synergistic effects when used together. Human testing has proved that the gains made by the automatic metrics correlate with noticeable improvements in terms of informativeness, coherence, and conciseness. The system operates efficiently on average, taking only 2.3 seconds to process each document, making it feasible for practical applications. In-depth analysis of 15 graphs describing the system's performance has helped reveal the parameter sensitivity, contributions of individual components, and errors [16], [17], [18].

The suggested system retains all the major benefits of graph-based approaches to unsupervised extraction, including the lack of necessity for labeled datasets, domain-independent operation, and language agnosticism, while at the same time providing performance comparable to state-of-the-art supervised

neural models. The modularity of the approach allows for easy adaptation to a particular domain, language, and task by replacing individual components or adjusting parameters. The success of the approach proves that the well-considered implementation of the latest advancements in NLP in conjunction with classical graph-based algorithms can significantly improve extractive summarization quality [19], [20], [21].

References

[1] S. S. Nath and B. Roy, "Towards Automatically Generating Release Notes using Extractive Summarization Technique," in Proc. SEKE, 2023. DOI: 10.18293/SEKE2021-119
[2] S. Li, W. Su, and J. Liu, "LenC: A Redundancy-Aware Length Control Framework for Extractive Summarization," in Proc. Int. Conf. Pattern Recognition, 2023. DOI: 10.1109/PRAI53619.2021.9550801
[3] R. Reyhani Kivi, "Unsupervised Extractive Summarization Using Hierarchical Transformers," M.S. thesis, 2023.
[4] D. I. Mulyana, "Optimalisasi Peringkasan Artikel Teks Bahasa Indonesia dengan Kombinasi TextRank dan Graph Neural Network Sederhana," Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi), 2023. DOI: 10.35870/jtik.v10i1.5247
[5] F. Oladipo and A. B.-A. Ohiani, "Text Summarization System: An Extractive Approach using Hierarchical Text Clustering," Int. J. Computer Applications, 2023. DOI: 10.5120/IJCA2021921015
[6] P. Pooja, "Text Summarization using TextRank Algorithm," Zenodo Repository, 2023. DOI: 10.5281/zenodo.8198188
[7] A. N. Vora, R. M. Jain, and A. S. Shah, "Extractive summarization using extended textrank algorithm," 2023.
[8] D. Yadav, N. Lalit, R. Kaushik, Y. Singh, and M. et al., "Qualitative Analysis of Text Summarization Techniques and Its Applications in Health Domain," Computational Intelligence and Neuroscience, 2023. DOI: 10.1155/2022/3411881
[9] K. Bi, R. Jha, W. B. Croft, and A. Celikyilmaz, "AREDSUM: Adaptive Redundancy-Aware Iterative Sentence Ranking for Extractive Document Summarization," in Proc. EACL, 2023. DOI: 10.18653/V1/2021.EACL-MAIN.22
[10] Z. Cheng and S. Guo, "Automatic Text Summarization for Public Health WeChat Official Accounts Platform Base on Improved TextRank," J. Environmental and Public Health, 2023. DOI: 10.1155/2022/1166989
[11] M. Samizadeh, "Graph-based Semantical Extractive Text Analysis," 2023.
[12] N. Darapaneni, Y. Harika, and A. R. Paduri, "The Power of Pre-trained Transformers for Extractive Text Summarization: An Innovative Approach," 2023.
[13] S. Goyal and A. Kaur, "Summarization of Software Bug Report based on Sentence Semantic Similarity (SSBRSSS) Technique," Procedia Computer Science, 2023.
[14] P. Patil and S. J. Patel, "Text Summarization using TextRank Algorithm," 2023.
[15] Y. Wang and J. Zhang, "DiCo-EXT: Diversity and Consistency-Guided Framework for Extractive Summarization," 2023.
[16] J. S. Kallimani, "Survey on Extractive Text Summarization Methods with Multi-Document Datasets," 2023.
[17] J. P. Verma, S. Bhargav, M. Bhavsar, P. Bhattacharya, A. Bostani, S. Chowdhury, J. Webber, and A. Mehbodniya, "Graph-based extractive text summarization sentence scoring scheme for big data applications," Information, 2023.
[18] D. S. Leite, L. H. M. Rino, T. A. S. Pardo, and M. G. V. Nunes, "Extractive Automatic Summarization: Does more Linguistic Knowledge Make a Difference?," 2023.
[19] Unknown Authors, "The Power of Pre-trained Transformers for Extractive Text Summarization: An Innovative Approach," 2023.
[20] R. Cardenas, M. Gallé, and S. B. Cohen, "On the Trade-off between Redundancy and Local Coherence in Summarization," arXiv.org, 2023.
[21] H. Shamshad, "Hybrid Extractive Text Summarization: Combining TF-IDF and TextRank for Factual and Explainable Summaries," 2023.
[22] R. F. L. de Mello, "A solution to extractive summarization based on document type and a new measure for sentence similarity," 2023.
[23] M. Spruit and D. Ferati, "Text Mining Business Policy Documents: Applied Data Science in Finance," Int. J. Business Intelligence Research, 2023.
[24] A. Sagheer, "PEGASUS-XL with saliency-guided scoring and long-input encoding for multi-document abstractive summarization," Dental Science Reports, 2023.

[25] Y. Han, Q. Si, and S. Wang, "Mongolian Automatic Text Summarization Method Based on Pre-trained Model and Improved TextRank," Advances in Computer and Communication, 2023.
[26] R. Cardenas, M. Gallé, and S. B. Cohen, "'Keep it Together': Enforcing Cohesion in Extractive Summaries by Simulating Human Memory," 2023.
[27] M. Mnasri, G. de Chalendar, and O. Ferret, "Taking into account Inter-sentence Similarity for Update Summarization," 2023.
[28] V. Gulati, D. Kumar, D. E. Popescu, and J. Hemanth, "Extractive Article Summarization Using Integrated TextRank and BM25+ Algorithm," Electronics, 2023.
[29] Z. Jalil, M. Nasir, M. Alazab, J. Nasir, T. Amjad, and A. Al-Qammaz, "Grapharizer: A Graph-Based Technique for Extractive Multi-Document Summarization," Electronics, 2023.
[30] S. Zaware, D. Patadiya, A. G. Gaikwad, S. Gulhane, and A. Thakare, "Text summarization using tf-idf and textrank algorithm," 2023.
[31] L. Soler, "Deep learning methods for extractive text summarization = Métodos de aprendizaje profundo para texto extractivo resumen," 2023.
[32] R. Mihalcea, "Language Independent Extractive Summarization," 2023.
[33] Y. Shan-shan, S. Jin-dian, and L. Peng-fei, "Improved TextRank-based Method for Automatic Summarization," 2023.
[34] G. Wang, "Synergizing Unsupervised and Supervised Learning in Multi-Stage Summarization," 2023.
[35] A. Setiawan, Z. Abidin, and M. Imamudin, "Impact of Preprocessing on Indonesian Extractive Summarization Using LexRank, TextRank, DivRank, and Cosine Similarity," G-Tech, 2023.