Optimizing Text Summarization with Sentence Clustering and Natural Language Processing

Zahir Edress*, Yasin Ortakci
Department of Computer Engineering, Karabuk University, Karabuk, Turkey

***Abstract*—Text summarization is an important task in natural language processing (NLP), with significant implications for information retrieval and content management. Traditional summarization methods often struggle with issues like redundancy, loss of key information, and inability to capture the underlying semantic structure of the text. This paper addresses these challenges by presenting an advanced approach to extractive summarization, which integrates clustering-based sentence selection with the BART model.The proposed method tackles the problem of redundancy by using Term Frequency-Inverse Document Frequency (TF-IDF) for feature extraction, followed by K-means clustering to group similar sentences. This clustering step is designed to reduce redundancy by ensuring that each cluster represents a distinct theme or topic. Representative sentences are then selected from these clusters based on their cosine similarity to a user query, which helps in retaining the most relevant information. These selected sentences are then fed into the BART model to generate the final abstractive summary. This combination of extractive and abstractive techniques addresses the common problem of information loss, ensuring that the summary is both comprehensive and coherent. The approach is evaluated using the CNN/DailyMail and XSum datasets, which are widely recognized benchmarks in the summarization domain. Results assessed through ROUGE metrics demonstrate that the proposed model substantially improves summarization quality compared to existing benchmarks.**

Keywords*—Abstractive summarization; extractive summarization; sentence clustering; language understanding; information retrieval

I. Introduction

Text summarization task aims to generate a concise and coherent summary from a given text while preserving its key information and meaning[1]. The challenge lies in effectively capturing the essence of the original content, which can vary widely in length, complexity, and structure. In the realm of text summarization, there are two primary approaches [2]: extractive and abstractive summarization, extractive summarization involves selecting and compiling sentences or phrases directly from the source text to form a summary. This approach relies on identifying the most important segments of the text, but may lack coherence and fluidity in the final output. On the other hand, abstractive summarization involves generating new sentences that convey the core ideas of the text [3], often resulting in more human-like and fluent summaries. However, this approach requires advanced techniques to ensure that the generated summary is both accurate and contextually relevant [4].

Extractive summarization is the most widely utilized and rapidly developed approach [5]. Despite its advancements, extractive summarization faces challenges, particularly in initial center selection and redundancy issues. In articles with complex sentence structures, the summarized sentences often exhibit high redundancy. This occurs because the extracted sentences may share significant semantic similarity, leading to repetitive content in the summary. Consequently, even if the summary includes relevant concepts, it may contain redundant information due to the repetition of similar concepts. This redundancy can make the summary unnecessarily lengthy and less concise. To solve the above problems this paper explores an enhanced summarization approach that combines clustering-based extractive methods with the BART model. the methodology starts by employing TF-IDF to evaluate the significance of each sentence within the original text and subsequently apply K-means clustering to group similar sentences into clusters. This process helps in identifying thematic groups within the text, which reduces redundancy and ensures a more diverse representation of content. For each cluster, representative sentences are selected based on cosine similarity to a query sentence. By selecting sentences that closely match this query, key information from each cluster is accurately captured, the selected sentences are then used to form a context for the BART model, which generates the final summary.

The proposed approach is evaluated using several well-established benchmark datasets, including CNN/DailyMail and XSum. Performance is assessed using ROUGE metrics, and the experimental results indicate that the model achieves superior ROUGE scores compared to other benchmark summarization methods. Specifically, the approach demonstrates significant improvements in terms of informativeness and coherence, reflecting a more accurate and fluid representation of the original text.

The main contributions of this work are summarized as follows:

- A novel framework for text summarization is proposed that enhances the coherence and human-likeness of the generated summaries.
- An effective sentence sampling strategy is introduced within the rewrite model, which significantly improves the quality and relevance of the summaries.
- The extensive experiments demonstrate that the proposed model consistently outperforms existing state-of-the-art baselines in text summarization.

The rest of the paper is organized as follows. Section II reviews related work of this paper. Section III gives a detailed explanation of the methodology. Section IV describes the experimental setups, which is followed by the experimental results in Section V. The paper is concluded in Section VI.

*Corresponding author

II. Related Work

Automatic Text summarization (ATS) has been extensively researched, with various methods proposed to address its challenges [6]. Early work in extractive summarization focused on approaches such as frequency-based methods, graph-based algorithms like Text Rank, and machine learning techniques [7]. These methods generally perform well in selecting salient sentences but often struggle with generating coherent summaries. Recent advancements in abstractive summarization have been driven by transformer-based models. Models like BERT (Bidirectional Encoder Representations from Transformers) and its variants have achieved state-of-the-art performance on summarization tasks by capturing deep contextual information [8]. BART a more recent model, combines the strengths of both bidirectional and autoregressive transformers, making it particularly effective for text generation tasks [1]. This section provides a summary of relevant research that informs the hybrid summarization approach, which combines extractive and abstractive techniques.

A. Extractive Text Summarization

The extractive summarization method involves selecting key sentences and keywords directly from the original text [9], It works by scoring sentences based on their importance, with higher scores indicating more significant sentences [10]. The summary is created by sequentially choosing the highest-scoring sentences from the text [11]. Liu et al. [7] enhanced extractive summarization by combining the improved TextRank algorithm with K-means clustering, This approach utilizes the BM25 model to compute sentence similarity and derive TR scores which are used to select initial cluster centers and reduce redundancy. Mohsen et al. [12] developed a hierarchical self-attentive neural network model that integrates reinforcement and supervised learning to rank sentences by directly optimizing the ROUGE metric. The model leverages hierarchical self-attention to create document and sentence embeddings that capture the document's structure, enhancing feature representation and summarization quality.

B. Abstractive Text Summarization

The abstractive summarization method creates summaries by generating new words and phrases, rather than directly extracting from the original text [13]. It uses natural language understanding to analyze the grammar and semantics of the document [14], allowing it to convey the main ideas in a newly formulated text [15]. Jain et al. [16] developed a specialized dataset for abstractive summarization to increase the number of training samples while maintaining manageable lengths for each sample. Bahrainian et al. [17] developed CATS an advanced abstractive neural summarization model that enhances the traditional sequence-to-sequence approach by introducing a mechanism for controlling the latent topic distribution of the generated summaries Su et al. [18] proposed a two-stage method for variable-length abstractive summarization using a text segmentation module and a Transformer-based summarization model. It first segments the text using BERT and BiLSTM, then applies a two-stage training approach with BERTSUM for document and segment summarization.

C. Hybrid Text Summarization

The hybrid summarization method combines both extractive and abstractive techniques to leverage the strengths of each approach. Initially, an extractive summary is generated using extraction models or trainers [8]. This summary is then refined through semantic analysis and rephrasing to produce an abstractive summary [19]. This dual approach aims to create a more comprehensive summary by integrating the benefits of both methods [20]. Recent studies have increasingly focused on combining extractive and abstractive methods [21]. For instance, using extractive summaries as inputs for abstractive models has demonstrated improved performance [22], highlighting the efficacy of this integrated approach in enhancing summary quality. Zhang et al. [1] proposed framework as Extract-then-Abstract approach, an initial extractive summarization model is used to identify and select relevant sentences from the text. This extractive summary is then refined using an abstractive model to generate summary. Morozovskii and Ramanna [23] proposed supervised learning model employs a hybrid approach that integrates both extractive and abstractive elements to enhance the inclusion of crucial information in news summaries. Wang et al. [13] introduced the Topic-Injected Bidirectional Encoder Representations from Transformers (TP-BERT) by incorporates contrastive learning during training.

The previous studies highlights various methodologies and advancements in text summarization. Table I summarizes different studies, detailing the year of the study, the methods employed, key contributions, and the limitations encountered. the proposed approach introduces a novel enhancement by integrating clustering-based sentence selection with the BART model. Unlike traditional extractive methods that select sentences based on frequency or similarity alone, the method utilizes TF-IDF for feature extraction and K-means clustering to group similar sentences. This clustering process identifies thematic groups within the text, reducing redundancy and ensuring a more diverse representation of content. By selecting representative sentences from each cluster based on cosine similarity to a query sentence, the approach effectively captures essential information while minimizing redundancy. This method addresses the limitations of improved TextRank, which may not fully achieve thematic coherence. Moreover, the integration with BART enhances the fluency and coherence of the final summary, leading to more contextually accurate and readable outputs.

III. Methodology

Our proposed summarization approach integrates extractive and abstractive techniques to enhance the relevance and coherence of generated summaries. The methodology is designed to capture key information from the source text while producing fluent and contextually accurate summaries. The process involves data preprocessing, vectorization, clustering-based sentence selection, representative sentence extraction, and abstractive summarization using BART as shown in Fig. 1.

A. Data Preprocessing

The preprocessing steps included Articles were retrieved from the dataset based on user-defined keywords using a

TABLE I. SUMMARY OF RELATED WORK IN TEXT SUMMARIZATION

Study	Year	Method	Key Contributions	Limitations
Wang et al [13]	2024	Integrates topic words into sentences and uses contrastive learning during training	Improving semantic consistency and summarization quality	Incorporating topic words and contrastive learning may add complexity to the model, potentially increasing computational requirements.
Jain et al [24]	2024	Generating additional training samples by creating multiple extractive summaries	Handling long documents	Potential information loss from segmenting documents into 512-token
Liu et al [7]	2024	Employs an improved BM25 model to compute BM25 similarity between sentences.	Reduce sentence redundancy	The method does not account for the position of sentences in the final summary, which might affect the coherence and readability of the summary
Morozovskii and Ramanna [23]	2023	Proposes an enhancement to the transformer model's attention mechanism by integrating frequency information for each word to better handle rare words.	Enhanced Attention Mechanism	The model may include less relevant information due to its distribution of attention across words
Zhang et al [1]	2023	Implements sentence sampling strategy to create parallel data for the rewrite model without manual annotation	Develops a pipeline using topic modeling to select the most relevant sentences	The extractive stage might still result in summaries that lack smooth transitions between sentences
Bahrainian et al[17]	2022	Neural sequence-to-sequence model with encoder-decoder and topical attention	Introduces customizable abstractive summarization using topic modeling.	Need for comparison with transformer-based models.
Su et al [18]	2020	Two-stage Transformer-based summarization model with text segmentation	Employs BERT and LSTM for segmentation and uses collaborative training.	The integrating of multiple models and collaborative training might imply potential challenges in practical implementation.
Mohsen et al [12]	2020	Hierarchical self-attentive reinforced neural network-based summarization model	Combines reinforcement and supervised learning to optimize the ROUGE evaluation metric directly.	The complexity of the model may lead to increased computational requirements.

search function that scans through the dataset and returns the top results. This search is limited to a maximum of five articles to maintain manageability. Each article was segmented into sentences using NLTK's "sent tokenize" function to facilitate clustering and similarity computations [25]. Sentence segmentation enables us to process and analyze each sentence independently [26]. Further preprocessing included stop words removal, which involved eliminating common but less informative words (e.g. "the", "and") to focus on more meaningful terms [27]. Also, word alignment process is to remove punctuation and numbers from each sentence. Finally, Lemmatization was applied to reduce words to their base or root forms (e.g. transforming "running" to "run") [28], Fig. 2 summarizes these steps.

B. Vectorization

In this step, the Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer is employed to convert sentences into numerical feature vectors, this process begins with applying the vectorizer to the preprocessed sentences, transforming them into a high-dimensional space where each dimension corresponds to a term in the vocabulary. The (TF-IDF) model assigns weights to terms based on their importance within each sentence relative to the entire corpus. By capturing both the frequency of terms within individual sentences (Term Frequency) and their significance across the corpus, the TF-IDF vectorizer produces feature vectors that effectively represent the content and context of each sentence.

C. Clustering Process

A clustering-based approach was employed to identify representative sentences [7]. Specifically, the K-means algorithm was used to cluster sentences into a predefined number of clusters. K-means partitions sentences based on their vector representations [1], grouping similar sentences together [29]. the elbow method was utilized to determine the optimal number of clusters as illustrated in Fig. 3. This method involves plotting the within-cluster sum of squares (inertia) against the number of clusters [30]. Inertia measures the sum of squared distances between each point and its assigned cluster center, reflecting the compactness of the clusters [31]. the K-means algorithm was executed across a range of cluster counts (from 1 to 10) and computed the inertia for each count.The resulting plot displays the number of clusters on the x-axis and the corresponding inertia on the y-axis. The "elbow" of the plot, where the rate of decrease in inertia slows down, indicates the optimal number of clusters. This point represents a balance between minimizing inertia and avoiding overfitting by selecting too many clusters [32]. choosing the optimal number of clusters affects the thematic diversity and coherence of the summary. Fewer clusters may lead to excessive redundancy, while more clusters can cause over-segmentation and loss of crucial information. The elbow method helps in finding a balance that improves summary quality by ensuring each

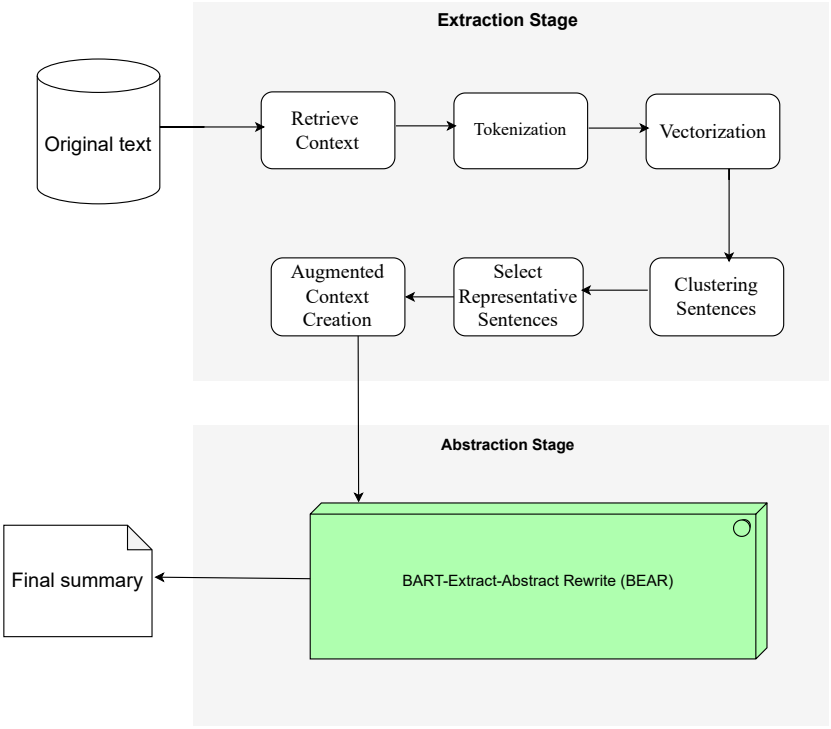

Fig. 1. The general structure of the proposed model.

cluster represents a distinct theme or topic, which enhances the effectiveness and the summarization results [33].

D. Selecting Representative Sentences

To refine the selection of representative sentences from each cluster, cosine similarity was calculated between a query sentence and each sentence within a cluster. The query was constructed from the first few sentences of the article to effectively represent the main context. For each cluster, the sentence with the highest cosine similarity to the query was selected as outlined in the Algorithm 1. cosine similarity was chosen because it effectively quantifies the similarity between sentences based on their term frequency representation [34], this metric is particularly useful for identifying sentences that closely match the thematic context of the query sentence [35], ensuring that the representative sentences are contextually relevant.

E. Summarization with BART

The BART (Bidirectional and Auto-Regressive Transformers) model, specifically the pre-trained facebook/bart-large-cnn version was used for generating summaries due to its strong performance on summarization tasks [1]. The model was configured to produce summaries with a maximum length of 150 tokens and a minimum of 40 tokens. It was fed with an augmented context, which was created by concatenating representative sentences selected through a clustering process. This augmented context helped the model generate a more coherent and comprehensive final summary.

Algorithm 1: Clustering-Based Extractive-Then-Abstractive Summarization

Input Text document D, User query Q
Output Final summary S
Feature Extraction: Calculate the Term Frequency-Inverse Document Frequency (TF-IDF) for each sentence in document D Represent each sentence as a TF-IDF vector
Sentence Clustering: Apply K-means clustering to group the sentences into K clusters based on their TF-IDF vectors
Representative Sentence Selection: for *each cluster* **do**
Calculate the cosine similarity between each sentence and the user query Q Select the sentence with the highest cosine similarity within each cluster as the representative sentence
Input to BART Model: Combine the selected representative sentences into a coherent sequence Feed the sequence of selected sentences into the pre-trained BART model
Generate Final Summary: The BART model generates an abstractive summary S from the input sequence of representative sentences
return Final summary S

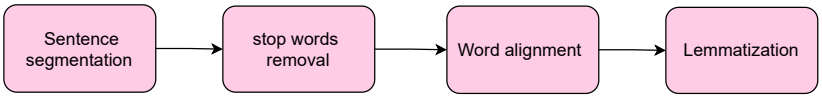

Fig. 2. Data preprocessing steps.

IV. Experiments

In this section, The experiments conducted to evaluate the effectiveness of the proposed summarization model are presented. These experiments are designed to assess the performance of various summarization methods across different datasets, with a focus on comparing baseline methods and the proposed approach.

A. Datasets

To evaluate the performance of the framework, it is assessed using two benchmark summarization datasets: CNN/DailyMail [36] and XSum [37], detailed statistical information for these datasets is provided in Table II.

1) The CNN/DailyMail: dataset is widely used for evaluating text summarization models. It consists of approximately 93,000 articles from CNN and 220,000 articles from the Daily Mail newspapers, totaling around 313,000 news articles [6]. Each article in the dataset is paired with multiple reference summaries, which are generally several sentences long, and are written by humans. The summaries are derived from the article highlights, typically found as bullet points making this dataset suitable for both extractive and abstractive summarization tasks.

2) The XSum: (Extreme Summarization) dataset is a benchmark dataset specifically designed for single-document abstractive summarization tasks. It consists of approximately 227,000 online articles from the BBC, covering a wide range of topics, including news, sports, and lifestyle. Each article is paired with a corresponding one-sentence summary, known as the "extreme summary", which captures the key point of the article. The one-sentence summaries are often highly abstractive, meaning they are not simply extracted sentences from the original text but are instead newly generated sentences that convey the main idea [7].

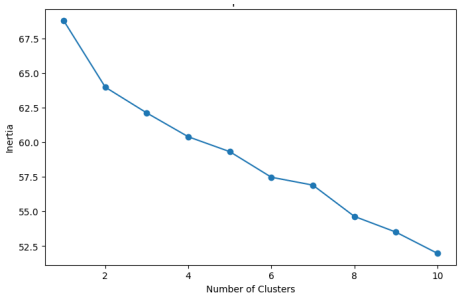

Fig. 3. Elbow method for determining the optimal number of clusters.

B. Baseline Methods

A range of high-performing models is selected for evaluation, relevant models from recent years as the baseline.

TABLE II. Statistics of the CNN/DailyMail and XSum Datasets

Dataset	#Docs	#Avg. sent.	#Avg. words	Domain
CNN/DailyMail	287,113	45.7	781	News
XSum	204,045	19.77	431	News

1) Lead-3: method simply selects the first three sentences from the document as the summary. It exploits the tendency of news articles to place the most important information at the beginning [38].

2) PacSum: (Parameterized Summarization) uses a heuristic-based approach to select sentences based on their importance, as determined by a parameterized model. It is known for its effectiveness in extracting informative sentences that contribute to high-quality summaries [39].
FAR (Focus-Aware Ranking) incorporates focus-aware ranking strategies to select the most relevant sentences. It improves upon simple extraction methods by considering the relevance and focus of the content [40].

3) SUMO: (Summarization with Meaningful Overlaps) focuses on identifying and using significant overlaps in sentence content to improve summary quality. It aims to balance informativeness and redundancy in the selected sentences [41].

4) PGNet: (Position-Aware Graph Network) incorporates position information and graph-based methods to rank sentences based on their importance in a given document. It improves upon traditional methods by leveraging positional information [16].

5) REFRESH: (REsponse-aware Frequency-based Summarization with Contextualized Highlights) uses response-aware frequency metrics to highlight key content. It adapts based on context to improve summary relevance [42].

6) SEQ: (Sequence-to-Sequence) models are based on sequence-to-sequence learning approaches that generate summaries by learning from large amounts of text. These models often use encoder-decoder architectures to produce coherent and contextually relevant summaries [43].
TED (Text Extensible Domain) leverages domain-specific knowledge to enhance summarization. By extending the text domain, TED models improve summary quality by incorporating additional contextual information [44].

7) CPSUM: (Soft) is an extractive summarization approach that employs a soft selection mechanism, focusing on selecting important sentences based on their relevance and contribution to the summary [13].

8) CPSUM: (Hard) differs from its soft counterpart by using a hard selection mechanism, which selects sentences more decisively without soft probability weighting [45].

9) BERTEXT: is an extractive summarization method that employs sequence labeling. This approach determines which sentences should be included in the summary by leveraging predictions made by a pre-trained BERT model [8].

10) BART: effectively captures context through its bidirectional encoder and produces high-quality summaries using its autoregressive decoder [1].

C. Evaluation

To evaluate the performance of the summarization approach, the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) metric was used to assess the quality of the generated summaries [19]. ROUGE measures the overlap between the generated summaries and reference summaries across several dimensions including [46]:

ROUGE-1 (R-1): Measures unigram overlap between the generated and reference summaries. This metric evaluates the proportion of unigrams (single words) in the generated summary that are present in the reference summaries [4].

ROUGE-2 (R-2): Assesses bigram overlap, focusing on pairs of consecutive words. This metric helps determine how well the generated summary captures the sequence of word pairs found in the reference summaries [13].

ROUGE-L (R-L): Evaluates the longest common subsequence between the generated and reference summaries [47]. The calculation of the ROUGE-N is shown in Eq. (1).

$$\text{ROUGE-N} = \frac{\sum_{s \in S} \sum_{n \in \text{N-grams}(s)} \text{count}_{\text{match}}(n)}{\sum_{s \in S} \sum_{n \in \text{N-grams}(s)} \text{count}_{\text{ref}}(n)} \quad (1)$$

where:

- S is the set of generated summaries.
- N-grams(s) represents the set of n-grams in summary s.
- $\text{count}_{\text{match}}(n)$ is the count of n-grams n that match between the generated summary and the reference summaries.
- $\text{count}_{\text{ref}}(n)$ is the count of n-grams n in the reference summaries.

D. Implementation

The implementation involved Libraries, the "transformers" library for BART was used, "datasets" for loading the datasets, "sklearn" for clustering and vectorization, and "nltk" for sentence tokenization. A Python script was developed to automate the entire process, from article retrieval and sentence clustering to summary generation and evaluation.

V. Results and Discussion

A. Results

The performance of the proposed BEAR model is evaluated using ROUGE metrics across various summarization models, as detailed in Table III. This table showcases the R-1, R-2, and R-L scores for different models on the CNN/DailyMail dataset, including the BEAR model. Notably, BEAR demonstrates competitive performance with R-1, R-2, and R-L scores of 48.15, 19.23, and 37.03, respectively, surpassing several baseline models.

To facilitate a visual interpretation of these results, Fig. 4 presents a heatmap of the ROUGE scores. This heatmap highlights the comparative performance of BEAR against other models, clearly illustrating that BEAR excels in both R-1 and R-2 metrics. This indicates its superior capability in capturing relevant information and ensuring coherence in summaries.

TABLE III. ROUGE Scores for Different Models on the CNN/DailyMail Dataset

Model	R-1	R-2	R-L
Lead 3	40.01	17.45	36.31
PacSum	40.37	17.92	36.62
FAR	40.42	17.95	36.67
SUMO	41.00	18.40	37.20
PGNet	39.50	17.30	36.40
REFRESH	41.30	18.40	37.50
SEQ	23.24	7.10	22.15
TED	38.73	16.84	36.15
BART	44.16	18.07	35.53
BEAR (proposed model)	**48.15**	**19.23**	**37.03**

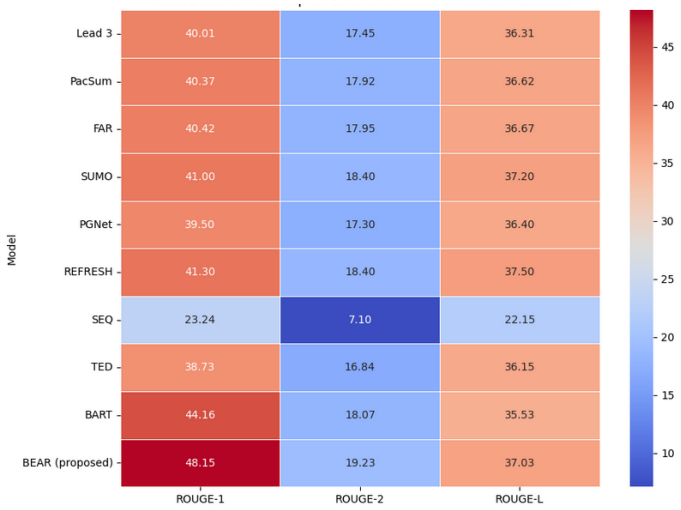

Fig. 4. Heatmap of ROUGE scores for different summarization models on the CNN/DailyMail dataset.

Table IV presents the ROUGE scores for various summarization models evaluated on the XSum dataset. Notably, the proposed BEAR model demonstrates impressive performance, achieving R-1, R-2, and R-L scores of 25.00, 8.57, and 19.44, respectively. Fig. 5 provides a visual representation of these results through a heatmap. In the heatmap, color gradients are used to illustrate the relative performance of each model, with darker shades indicating higher ROUGE scores. This visualization effectively highlights the strengths of the BEAR model compared to other models, underscoring its capability in generating high-quality summaries.

TABLE IV. ROUGE SCORES FOR VARIOUS SUMMARIZATION MODELS ON THE XSUM DATASET

Model	R-1	R-2	R-L
Lead 3	16.30	1.60	11.95
SEQ	20.11	5.23	16.15
CPSUM(Soft)	17.22	2.17	12.71
CPSUM(Hard)	17.29	2.18	12.73
BERTEXT	22.86	4.48	17.16
BEAR (proposed model)	**25.00**	**8.57**	**19.44**

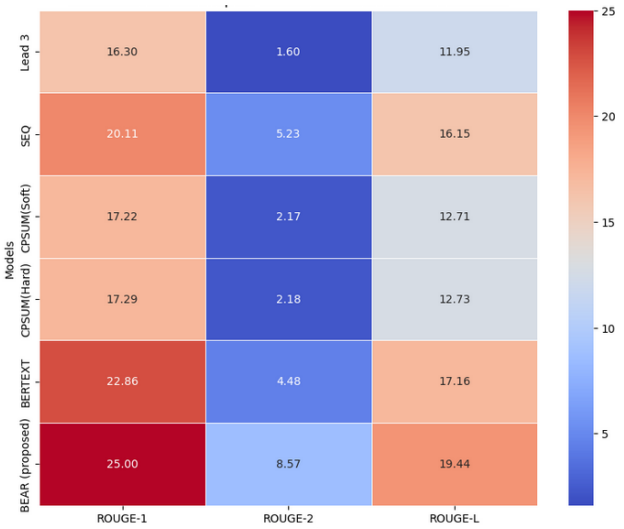

Fig. 5. Heatmap of ROUGE scores for various summarization models on the XSum dataset.

B. Comparative Analysis with Previous Research

To assess the relevance and competitiveness of the BEAR model, its results were compared with those of other state-of-the-art summarization approaches. Table V shows a detailed comparison of ROUGE scores across different models on the CNN/DailyMail and XSUM datasets, highlighting the BEAR model's superior performance.

1) CNN/DailyMail dataset: The BEAR model outperforms several baseline models such as Lead-3, PacSum, and REFRESH in terms of R-1 and R-2 scores, indicating its superior ability to capture essential content. For example, Lead-3 achieved an R-1 score of 40.01, while BEAR obtained 48.15, demonstrating significant improvements compared to BERTSUM, which has been widely used for summarization, BEAR's higher ROUGE scores underscore its effectiveness in integrating clustering-based sentence selection with abstractive summarization.

2) XSum dataset: The BEAR model outperforms models like BERTEXT and SEQ in both R-1 and R-2 scores. The comparative analysis of the proposed BEAR model against other summarization models on the XSum dataset demonstrates its superior performance across all three ROUGE metrics (R-1, R-2, and R-L) key observations include:

3) ROUGE-1 (R-1): BEAR achieved an R-1 score of 25.00, outperforming other models, such as SEQ (20.11) and BERTEXT (22.86), which are commonly used for summarization tasks. This higher score indicates that BEAR can better capture important words and phrases from the original text.

4) ROUGE-2 (R-2): With an R-2 score of 8.57, BEAR substantially surpasses the other models in capturing bigram-level co-occurrences between the generated summaries and the reference summaries. For instance, SEQ and BERTEXT which scored 5.23 and 4.48 respectively, lag behind in capturing contextual relationships that involve pairs of words. The improvement in R-2 reflects BEAR's enhanced ability to generate coherent summaries.

5) ROUGE-L (R-L): BEAR also leads in the ROUGE-L metric with a score of 19.44, demonstrating its strength in maintaining the longest common subsequence between the generated summaries and the references. This indicates that BEAR is more capable of producing summaries that retain the overall meaning and structure of the original content compared to other models like SEQ (16.15) and CPSUM (Soft) (12.71).

Overall, BEAR's results on the XSum dataset show that the model consistently outperforms the other methods across all metrics, validating its capability to generate high-quality summaries for summarization tasks.

C. Role of Clustering and Representative Sentences

Clustering organizes sentences into thematic groups based on their semantic similarity, utilizing TF-IDF and K-means clustering algorithms. This method structures the text into distinct clusters, each representing specific topics or aspects, which helps maintain context and relevance throughout the summarization process. By segmenting the text into these thematic clusters, redundancy is minimized and the focus is placed on selecting the most representative sentences. This reduces repetition and ensures a concise, relevant summary.

Once the clustering is complete, representative sentences from each cluster are selected and serve as the context for the BART model. The BART model then generates the final summary by integrating these representative sentences, this hybrid approach effectively bridges the gap between extractive and abstractive summarization techniques. It ensures that the summary not only captures key points but also maintains fluency and coherence, facilitating smooth transitions between different sections. The combination of clustering and BART helps to create a summary that is both contextually relevant and well-structured.

D. Limitations of the Present Study

While the approach aims to enhance coherence and relevance, there might still be instances where the generated summaries contain redundant or less relevant information if the clustering or similarity measures are not perfectly aligned with the summary objectives.

VI. CONCLUSION

In this paper, a novel framework for text summarization is presented that enhances the quality of generated summaries. The approach integrates a rewrite model with an innovative sentence sampling strategy, resulting in summaries that are not only more relevant but also of higher quality. A method for the rewrite model leverages an effective sentence sampling strategy, enhancing the selection of content for summarization.

TABLE V. STATE-OF-THE-ART COMPARISON FOR TEXT SUMMARIZATION

Model	CNN/DailyMail			XSUM		
	R-1	R-2	R-L	R-1	R-2	R-L
Lead 3	40.01	17.45	36.31	16.30	1.60	11.95
PacSum	40.37	17.92	36.62	-	-	-
FAR	40.42	17.95	36.67	-	-	-
SUMO	41.00	18.40	37.20	-	-	-
PGNet	39.50	17.30	36.40	-	-	-
REFRESH	41.30	18.40	37.50	-	-	-
SEQ	23.24	7.10	22.15	20.11	5.23	16.15
TED	38.73	16.84	36.15	-	-	-
BART	44.16	18.07	35.53	-	-	-
CPSUM (Soft)	-	-	-	17.22	2.17	12.71
CPSUM (Hard)	-	-	-	17.29	2.18	12.73
BERTEXT	-	-	-	22.86	4.48	17.16
BEAR (proposed model)	**48.15**	**19.23**	**37.03**	**25.00**	**8.57**	**19.44**

The framework consistently outperforms existing state-of-the-art baselines across various text summarization tasks. These results underscore the superiority of the approach in delivering more coherent and contextually relevant summaries. Future work will focus on several key areas to further enhance the summarization framework. First, the application of this approach to other languages and domains will be explored, expanding its scope and effectiveness. Additionally, methods to incorporate user feedback into the summarization process will be explored, enabling the generation of more personalized and user-centric summaries.

AUTHORS' CONTRIBUTIONS

Zahir Edrees: Conceptualization, Methodology, Software, Writing original draft, Validation, Resources, Visualization, review and editing.Yasin Ortakci: Conceptualization, Methodology, Project administration, Supervision, review and editing.

DATA AVAILABILITY STATEMENT

The datasets generated during and/or analyzed during the current study are available from the corresponding author upon reasonable request.

COMPETING INTERESTS

The authors have no competing interests to declare that are relevant to the content of this article.

CONFLICT OF INTERESTS

This manuscript has not been submitted to, nor is under review at, another journal or other publishing venue.

ETHICS AND INFORMED CONSENT FOR DATA USED

The research does not involve human participants and/or animals.

FUNDING AND ACKNOWLEDGMENT

This research received no external funding.

REFERENCES

[1] S. Zhang, R. Yang, X. Xiao, X. Yan, and B. Tang, "Effective and efficient pagerank-based positioning for graph visualization," *Proceedings of the ACM on Management of Data*, vol. 1, pp. 1–27, 5 2023.

[2] S. Gongid, Z. Zhu, J. Qi, C. Tong, Q. Lu, and W. Wu, "Improving extractive document summarization with sentence centrality," 2022. [Online]. Available: https://doi.org/10.1371/journal.pone.0268278.g001

[3] X. Zhang, F. Wei, and M. Zhou, "Hibert: Document level pre-training of hierarchical bidirectional transformers for document summarization," *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, pp. 5059–5069, 2019. [Online]. Available: https://aclanthology.org/P19-1499

[4] Y. Liu, P. Liu, D. Radev, and G. Neubig, "Brio: Bringing order to abstractive summarization," *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 2890–2903, 3 2022. [Online]. Available: https://arxiv.org/abs/2203.16804v1

[5] H. T. Kesgin and M. F. Amasyali, "Advancing nlp models with strategic text augmentation: A comprehensive study of augmentation methods and curriculum strategies," *Natural Language Processing Journal*, vol. 7, p. 100071, 6 2024.

[6] S. Pawar, H. M. Gururaj, and N. N. Chiplunar, "Text summarization using document and sentence clustering," vol. 215. Elsevier B.V., 2022, pp. 361–369.

[7] W. Liu, Y. Sun, B. Yu, H. Wang, Q. Peng, M. Hou, H. Guo, H. Wang, and C. Liu, "Automatic text summarization method based on improved textrank algorithm and k-means clustering," *Knowledge-Based Systems*, vol. 287, 3 2024.

[8] Y. Liu, "Fine-tune bert for extractive summarization." [Online]. Available: https://arxiv.org/abs/1908.08345

[9] S.-N. Vo, T.-T. Vo, and B. Le, "Interpretable extractive text summarization with meta-learning and bi-lstm: A study of meta learning and explainability techniques," *Expert Systems With Applications*, vol. 245, p. 123045, 2024. [Online]. Available: https://doi.org/10.1016/j.eswa.2023.123045

[10] M. E. Saleh, Y. M. Wazery, and A. A. Ali, "A systematic literature review of deep learning-based text summarization: Techniques, input representation, training strategies, mechanisms, datasets, evaluation, and challenges," *Expert Systems With Applications*,

vol. 252, p. 124153, 2024. [Online]. Available: https://doi.org/10.1016/j.eswa.2024.124153

[11] M. Zhong, P. Liu, Y. Chen, D. Wang, X. Qiu, and X. Huang, "Extractive summarization as text matching," pp. 6197–6208. [Online]. Available: https://github.

[12] F. Mohsen, J. Wang, and K. Al-Sabahi, "A hierarchical self-attentive neural extractive summarizer via reinforcement learning (hsasrl)," *Applied Intelligence*, vol. 50, pp. 2633–2646, 9 2020.

[13] Y. Wang, J. Zhang, Z. Yang, B. Wang, J. Jin, and Y. Liu, "Improving extractive summarization with semantic enhancement through topic-injection based bert model," *Information Processing and Management*, vol. 61, 5 2024.

[14] I. Benedetto, M. L. Quatra, L. Cagliero, L. Vassio, and M. Trevisan, "including those for text and data mining, ai training, and similar technologies. tasp: Topic-based abstractive summarization of facebook text posts," *Expert Systems With Applications*, vol. 255, p. 124567, 2024. [Online]. Available: https://doi.org/10.1016/j.eswa.2024.124567

[15] H. Shakil, A. Farooq, and J. Kalita, "To appear in: Neurocomputing," 2024. [Online]. Available: https://doi.org/10.1016/j.neucom.2024.128255.

[16] J. You, R. Ying, and J. Leskovec, "Position-aware graph neural networks." [Online]. Available: http://snap.stanford.

[17] S. A. Bahrainian, G. Zerveas, F. Crestani, and C. Eickhoff, "Cats: Customizable abstractive topic-based summarization," *ACM Transactions on Information Systems*, vol. 40, 1 2022.

[18] M.-H. Su, C.-H. Wu, and H.-T. Cheng, "A two-stage transformer-based approach for variable-length abstractive summarization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2061–2072, 2020.

[19] M. Zhang, C. Li, M. Wan, X. Zhang, and Q. Zhao, "Rouge-sem: Better evaluation of summarization using rouge combined with semantics," *Expert Systems With Applications*, vol. 237, 2024. [Online]. Available: https://doi.org/10.1016/j.eswa.2023.121364

[20] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, L. Màrquez, C. Callison-Burch, and J. Su, Eds. Lisbon, Portugal: Association for Computational Linguistics, Sep. 2015, pp. 379–389. [Online]. Available: https://aclanthology.org/D15-1044

[21] A. See, P. J. Liu, and C. D. Manning, "Get to the point: Summarization with pointer-generator networks," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, R. Barzilay and M.-Y. Kan, Eds. Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 1073–1083. [Online]. Available: https://aclanthology.org/P17-1099

[22] R. Nallapati, B. Zhou, C. N. dos santos, C. Gulcehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence rnns and beyond," 2 2016. [Online]. Available: http://arxiv.org/abs/1602.06023

[23] D. Morozovskii and S. Ramanna, "Rare words in text summarization," *Natural Language Processing Journal*, vol. 3, p. 100014, 6 2023.

[24] D. Jain, M. D. Borah, and A. Biswas, "Summarization of lengthy legal documents via abstractive dataset building: An extract-then-assign approach," *Expert Systems with Applications*, vol. 237, 3 2024.

[25] W.-W. Qiu, H.-T. Yu, C.-H. Tsai, D. Zhu, M.-H. Chen, and H. J. Kim, "Understanding the value of host-guest intimacy behind online reviews of airbnb," *International Journal of Hospitality Management*, vol. 115, p. 103599, 2023. [Online]. Available: https://doi.org/10.1016/j.ijhm.2023.103599

[26] "Artificial intelligence approach for detection and classification of depression among refugees in selected diasporic novels — enhanced reader."

[27] P. Radhakrishnan and G. SenthilKumar, "Stab: An enhanced abstractive text summarization employing stacked bi-gru with the attention cnn approach," *SN Computer Science*, vol. 5, 8 2024.

[28] M. Kirmani, G. Kour, M. Mohd, N. Sheikh, D. A. Khan, Z. Maqbool, M. A. Wani, and A. H. Wani, "Biomedical semantic text summarizer," *BMC Bioinformatics*, vol. 25, 12 2024.

[29] D. Parnes and A. Gormus, "Prescreening bank failures with k-means clustering: Pros and cons," *International Review of Financial Analysis*, vol. 93, p. 103222, 2024. [Online]. Available: https://doi.org/10.1016/j.irfa.2024.103222

[30] I. Trabelsi, R. Hérault, H. Baillet, R. Thouvarecq, L. Seifert, and G. Gasso, "Identifying patterns in trunk/head/elbow changes of riders and non-riders: A cluster analysis approach," *Computers in Biology and Medicine*, vol. 143, p. 105193, 2022. [Online]. Available: https://doi.org/10.1016/j.compbiomed.2021.105193

[31] W. A. Prastyabudi, A. N. Alifah, and A. Nurdin, "Sciencedirect peer-review under responsibility of the scientific committee of the seventh information systems international conference segmenting the higher education market: An analysis of admissions data using k-means clustering," 2023. [Online]. Available: www.sciencedirect.comwww.sciencedirect.com

[32] "Study of anisotropy in polydispersed 2d micro and nanocomposites by elbow and k-means clustering methods — enhanced reader."

[33] N. Rylko, M. Stawiarz, P. Kurtyka, and V. Mityushev, "Study of anisotropy in polydispersed 2d micro and nanocomposites by elbow and k-means clustering methods," *Acta Materialia*, vol. 276, p. 120116, 2024.

[34] T. P. Rinjeni, A. Indriawan, and A. Rakhmawati, "Sciencedirect peer-review under responsibility of the scientific committee of the seventh information systems international conference matching scientific article titles using cosine similarity and jaccard similarity algorithm," *Procedia Computer Science*, vol. 234, pp. 553–560, 2024.

[35] T. Nakagawa, Y. Sanada, H. Waida, Y. Zhang, Y. Wada, O. Takanashi, T. Yamada, and T. Kanamori, "Denoising cosine similarity: A theory-driven approach for efficient representation learning," *Neural Networks*, vol. 169, pp. 893–6080, 2024. [Online]. Available: https://doi.org/10.1016/j.neunet.2023.10.027

[36] K. Moritz, H. Tomas, K. Kocisky, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, P. Blun-

som, and G. Deepmind, "Teaching machines to read and comprehend." [Online]. Available: http://www.github.com/deepmind/rc-data/

[37] S. Narayan, S. B. Cohen, and M. Lapata, "Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization," pp. 1797–1807.

[38] R. Barzuay, K. R. Mckeown, and M. Elhadad, "Information fusion in the context of multi-document summarization."

[39] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," *Annual Meeting of the Association for Computational Linguistics*, 2004.

[40] G. Erkan and D. R. Radev, "Lexrank: Graph-based lexical centrality as salience in text summarization," *Journal Of Artificial Intelligence Research*, vol. 22, pp. 457–479, 9 2011. [Online]. Available: http://arxiv.org/abs/1109.2128 http://dx.doi.org/10.1613/jair.1523

[41] U. Jayasankar, V. Thirumal, and D. Ponnurangam, "A survey on data compression techniques: From the perspective of data quality, coding schemes, data type and applications," *Journal of King Saud University - Computer and Information Sciences*, vol. 33, pp. 119–140, 2 2021.

[42] S. Narayan, N. Papasarantopoulos, S. B. Cohen, and M. Lapata, "Neural extractive summarization with side information." [Online]. Available: www.aaai.org

[43] I. S. Google, O. V. Google, and Q. V. L. Google, "Sequence to sequence learning with neural networks," 2014.

[44] C. Li, Q. Li, P. V. Mieghem, H. E. Stanley, and H. Wang, "Correlation between centrality metrics and their application to the opinion model," *The European Physical Journal B*, vol. 88, p. 65, 12 2015.

[45] A. Onan and H. A. Alhumyani, "Fuzzytp-bert: Enhancing extractive text summarization with fuzzy topic modeling and transformer networks," *Journal of King Saud University-Computer and Information Sciences*, vol. 36, p. 102080, 2024. [Online]. Available: https://doi.org/10.1016/j.jksuci.2024.102080

[46] G. Sharma, D. Sharma, and M. Sasikumar, "Summarizing long scientific documents through hierarchical structure extraction," *Natural Language Processing Journal*, vol. 8, p. 100080, 2024. [Online]. Available: https://doi.org/10.1016/j.nlp.2024.100080

[47] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 74–81. [Online]. Available: https://aclanthology.org/W04-1013