ISSN 1840-4855
e-ISSN 2233-0046

Original scientific article
http://dx.doi.org/10.70102/afts.2025.1834.084

A PMI-DRIVEN APPROACH WITH CONVENTIONAL BERT FOR OPTIMIZING TEXT SUMMARIZATION

Dr.R. Ramesh[1], Dr.N. Subalakshmi[2], Dr.S. Selvarani[3], Dr.K. Kavitha[4*], Dr.M. Jeyakarthic[5]

[1]Assistant Professor, Department of Computer Applications, Thanthai Periyar Government Arts & Sciences College, Trichy, Tamil Nadu, India.
e-mail: rameshau04@gmail.com, orcid: https://orcid.org/0000-0002-1121-2223
[2]Assistant Professor, Department of Computer and Information Science, Annamalai University, Chidambaram, Tamil Nadu, India. e-mail: subhaabi12@gmail.com,
orcid: https://orcid.org/0000-0003-1139-6154
[3]Assistant Professor, Department of Computer Science, Alagappa Government Arts College, Karaikudi, Tamil Nadu, India. e-mail: samyselvaa@gmail.com,
orcid: https://orcid.org/0000-0001-6483-5654
[4]Assistant Professor (Selection Grade), Department of Electrical & Electronics Engineering, Annamalai University, Annamalai Nagar, Chidambaram, Tamil Nadu, India. e-mail: kavitha_au04@yahoo.com, orcid: https://orcid.org/0000-0003-4492-8997*
[5]Assistant Professor, Department of Computer and Information Science, Annamalai University, Annamalai Nagar, Chidambaram, Tamil Nadu, India.
e-mail: jeya_karthic@yahoo.com, orcid: https://orcid.org/0000-0001-6822-6004

Received: August 11, 2025; Revised: September 27, 2025; Accepted: November 06, 2025; Published: December 30, 2025

SUMMARY

Text summarization plays a crucial role in natural language processing by condensing large volumes of textual information into concise and meaningful summaries. With the rapid growth of digital content, existing summarization approaches often struggle to balance contextual understanding and semantic relevance. This paper presents a PMI-driven BERT-based text summarization framework that integrates Pointwise Mutual Information (PMI) as a statistical pre-processing mechanism with a fine-tuned Conventional BERT model to enhance summary quality. PMI is employed to identify and rank semantically significant terms based on co-occurrence patterns, enabling effective keyword and phrase prioritization before summarization. The ranked textual representation is then processed using a summarization-specific decoder layer added on top of the BERT encoder to generate coherent and context-aware summaries. The proposed framework is evaluated on the CNN/Daily Mail dataset comprising over 300,000 news articles, using ROUGE-1, ROUGE-2, and ROUGE-L metrics for performance assessment. Experimental results demonstrate that the proposed method achieves ROUGE-1, ROUGE-2, and ROUGE-L scores of 46.9, 27.61, and 45.68 respectively, outperforming baseline models such as Seq2Seq, Seq2Sick, and Prefix-Tuning by an average margin of 2–3%. The experiments were conducted using Python with the PyTorch deep learning framework on a CPU-based environment. The results indicate that PMI-based pre-processing significantly improves contextual relevance and semantic consistency in generated summaries. This framework demonstrates robustness and scalability, making it suitable for large-scale document summarization tasks.

Key words: *text summarization, pointwise mutual information, BERT, keyword extraction, rank terms, rouge.*

INTRODUCTION

The Internet is a broad environment where various web resources, such as websites and user feedback, news articles, blogs, and social media networks among others, constitute an enormous store of text information. The magnitude of this digital space and its variety, make it an unrivalled source of information, views and opinions. The dissemination of the online content has altered the means of gaining access, consuming, and contributing to the information [1]. The richness of user-generated material, instant news, and diversity of the opinions condensed in the web sources highlight the dynamics of this virtual world. Text summarization is a central operation included in the context of language processing and information retrieval, which transforms the long documents into brief and informative summaries of the necessary information [2]. In the age of information overload, when a great deal of literal information is produced on a daily basis, the ability to find the most important insights as quickly as possible is of utmost importance [3]. This has given rise to the recent explosion in the advancement and discovery of other text summarization methods [4]. The basic purpose of summarizing is to obtain the main idea and the main points of a manuscript so that people could understand what this manuscript is about not reading the whole text. Effective summarization does not only help in understanding of information [5]; it also helps in making fast decisions, improves information management systems as well as automating many operations [6].

In the constantly growing context of information search and processing, it has become a task to extract significant information of the enormous ocean of written content, which has led to the development of novel methods of text summarization [7]. Among them, the combination of PMI as a pre-processing step to the already developed BERT model [8] as a new and promising approach appears. The study provides a complex approach that capitalizes on the interplay between a term association strength measure that is based on PMI [9] and BERT to streamline the text summarization process. The optimization of the parameters of BERT normally pertains to fine-tuning of model parameters. To learn context representations of words, BERT learns on large text-data corpus [10]. The study explores the complicated interaction between the statistical metrics, denoted by PMI, and the most sophisticated skills of BERT, with the purpose of resulting in the improved efficiency and effectiveness of text summarization. The study through a stringent performance analysis throws light on the effectiveness of this integrated methodology which throws light on the possibility of making great progress in the field. When delving into the complexities of this PMI-based strategy together with Conventional BERT, we expect to achieve a subtle insight into the way the union between statistical analysis and sophisticated neural networks can result in the optimization of the most important process of text summarizing. The following are the primary contributions of the suggested method:

- The integration of PMI as a pre-processing step is a significant contribution.
- The combination of PMI-driven pre-processing and BERT-based summarization is designed to enhance the efficiency of the text summarization process.
- This ensures that the produced summaries maintain a high level of coherence and internment the crucial meaning of the input text.

The research work is organized as follows: section 2 discusses about the overall related works, section 3 presents the proposed model of Optimized Text Summarisation, section 4 discusses about the results along with comparison and finally work is concluded in section 5 with future directions.

RELATED WORKS

The purpose of text summarization is to create a shortened version of a given text while retaining the original material's meaning. It needs an automated method to assist with the existing depository of information [11]. This article [12] has a particular focus on the problem of legal text summarising since doing so is one of the most significant things that can be done in the legal arena. This article [13] gives a detailed literature assessment on the various sequence-to-sequence models for abstractive text summarising. Subsequently, these models were used for the purpose of abstract text summarization. The research presented in this article [14] blends lexical and neural rating algorithms for the purpose of retrieving case law. Recent work that has been done to pre-train Transformers with self-supervised goals on big text corpora has demonstrated natural language processing tasks such as text summarization [15]. Training and assessment are becoming more hampered by the data and metrics that are being utilised for a given job [16].

To exploit big pretrained language models for use in downstream activities, the de facto method that is currently in use is called fine-tuning. On the other hand, it changes every parameter of the language model, which means that a complete copy of the model has to be saved for each job [17]. In this study [18], a complete and consistent re-assessment of 14 automated evaluation criteria is performed by using the outputs of neural summarization models in conjunction with expert and crowd-sourced human comments. The majority of today's methods for generating text are based on autoregressive models and maximum likelihood estimates [27]. Due to a mismatch between the learning aim and the assessment measure, this paradigm generates samples that are diversified but of poor quality [19]. This article [20] provides a new self-supervised aim known as future n-gram prediction and a suggested n-stream self-attention mechanism. The purpose of the work [21] is to advance the understanding of how neural extractive summarization systems can benefit through the variety of model structures, transferable information, and learning structures. The use of neural sequence transduction approaches has been explored in the field of abstractive summarization, using datasets consisting of huge sets of document-summary pairs [22]. However, the majority of extant literature primarily focuses on addressing the image classification issue due to its characteristic of having a continuous input space and a limited output space [23].

Maximum likelihood estimation (MLE) is a widely used approach for training sequence-to-sequence models [26]. Nevertheless, in typical maximum likelihood estimation (MLE) training, the focus is on a word-level goal where the task involves estimating the subsequent word based on the preceding ground-truth partial phrase [24]. The present methodology places emphasis on the modelling of regional syntactic patterns, potentially overlooking the incorporation of extensive semantic structure [28]. The above research [25] is a new method that transforms the neural extractive summarization models.

Inference from Literature Survey

From the reviewed literature, it is observed that most text summarization approaches rely on sequence-to-sequence and transformer-based models with extensive fine-tuning. Although these methods capture strong contextual representations, they often neglect explicit statistical term association measures that can enhance semantic relevance. Moreover, heavy dependence on large-scale fine-tuning increases computational complexity and limits domain adaptability. Challenges related to long-document handling, keyword prioritization, and evaluation consistency also persist. To address these gaps, this research proposes a PMI-driven pre-processing strategy integrated with a conventional BERT-based summarization framework, aiming to improve summary relevance and coherence while maintaining computational efficiency.

PROPOSED MODEL

In this section, we propose Text summarization model which is optimized by Conventional BERT for generating summaries within length constraints. PMI is an effective pre-processing procedure to optimize the process of identifying significant terms and phrases to proceed with the real summarization via Conventional BERT. This novel method uses PMI to measure the quality of associations among terms in the input text, which gives a more insight into semantic associations. Figure 1 represents an Overall architecture of Text Summarization Model.

Ranking of the terms in the document is done using the calculated PMI scores. An optimized BERT model takes the processed text as the input with relevant terms or phrases or any other information derived through PMI. PMI-based pre-processing has a potential to improve the summarization process by bringing more insight or feature obtained out of the text, which helps the BERT model to capture the necessary information to create summarization more efficiently.

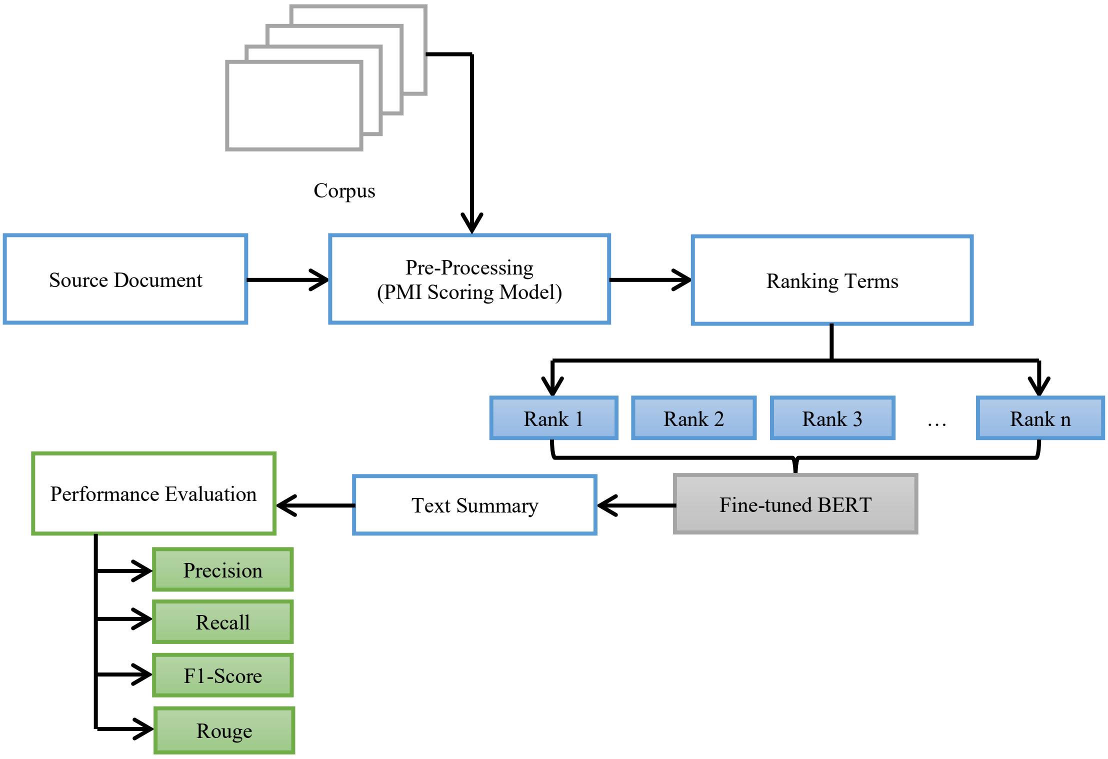

Figure 1. Architecture of text summarization model

Pre-processing

PMI scores between two terms of the corpus are computed in order to define strong relationships between the terms on the basis of their co-occurrence. The increased PMI scores imply that there is a greater possibility of co-occurrence which implies those terms are semantically connected. Ranking of terms by the PMI scores should be done so as to select the key phrases most likely to be of more importance during the process of summarization. According to the high PMI scores, the identified key phrases can be used as significant components that add some meaning to the whole meaning of the text. These critical phrases are then fed into the following summarization task and the objective is to extract the contextual relationship between terms and the ultimate output is a more contextually sensitive input. The PMI between two terms A and B is calculated using Equation (1).

$$PMI(A,B) = log_2\left(\frac{P(A,B)}{P(A)\cdot P(B)}\right) \quad (1)$$

Where: $P(A,B)$ is the probability of co-occurrence of terms A and B. $P(A)$ is the probability of occurrence of term A. $P(B)$ is the probability of occurrence of term B. In the context of text summarization, the probabilities are computed based on the frequency of occurrence of terms within the corpus. The probability of a term occurring in the document corpus is computed using normalized frequency statistics. Equation (2) defines the probability of term Aas the ratio of its frequency to the total number of terms in the corpus. Equation (3) similarly represents the probability of term Bbased on its occurrence frequency. The joint probability of terms $A$and Bco-occurring within the corpus is formulated in Equation (4) using their joint frequency normalized by the total number of corpus terms., while Equation (5) describes the corpus-level probability by aggregating term frequencies across all documents. Each equation is numbered separately and referenced individually along with its corresponding description to ensure clarity and precise interpretation.

1. Calculate Term Frequencies: $freq(A)$: No. of occurrences of term A in the corpus. $freq(B)$: No. of occurrences of term B in the corpus. $freq(A,B)$: Number of terms A and B in the corpus.
2. Calculate Probabilities:

$$P(A) = \frac{freq(A)}{Total\ number\ of\ terms\ in\ the\ corpus} \quad (2)$$

$$P(B) = \frac{freq(B)}{Total\ number\ of\ terms\ in\ the\ corpus} \quad (3)$$

$$P(A,B) = \frac{freq(A,B)}{Total\ number\ of\ terms\ in\ the\ corpus} \quad (4)$$

3. Calculate PMI:

$$PMI(A,B) = log_2\left(\frac{P(A,B)}{P(A) \cdot P(B)}\right) \quad (5)$$

This PMI score gives you a measure of the association between terms A and B. Higher PMI scores indicate a stronger association.

OPTIMIZED TEXT SUMMARIZATION

BERT is a transformer-based model that is famous in perceiving context in natural language operations. It comprises of an encoder architecture that extracts bidirectional contextual information on input sequences. A Summarization-Specific Layer (Decoder), which is placed on the top of BERT base model, is added to adapt it to text summarization. This extra layer is meant to fit the BERT with the capability of generating summaries as opposed to its defaulting task of classification. The text summary with Conventional BERT would mean fine-tuning the existing BERT model on a summarization-specific Layer as illustrated in figure 2. Summarization-Specific Layer transforms the representations received by the BERT encoder into the summary-specific representations. It may have a single layer or more layers like linear transformations, activation functions and may have more attention mechanisms that has specialized modules that are designed to perform the task of summarization. The Summarization-Specific Layer is a modification to the basic model of BERT that is fine-tuned on a dissimilar point's dataset involving summarization. During the fine-tuning, the model is trained to generate summary information, or representations containing significant information about the input text.

The adaptation of the BERT model to the summarization specific layer is performed through fine-tuning, whereby a training dataset comprising of input documents and associated summaries is used to train the fine-tuned BERT model. This is done to modify the weights of the model to enhance the ability to summarize. The trained model takes the input document, processes it through the modified BERT architecture, and produces summary representations, which are then post-processed to generate human-readable summaries. Text summarization using an optimized BERT with a Summarization-Specific Layer involves describing the operations and transformations occurring within pre-trained BERT model.

BERT, being a Transformer-based model, comprises many layers of self-attention techniques and feed-back neural networks. This can be represented in Equation (6):

$$BERT_{encoder}(input_{ids}) = Encoder(input_{ids}) \quad (6)$$

Where:

- $input_{ids}$ represents the tokenized input text.
- $Encoder(\cdot)$ represents the BERT encoder layers performing operations like self-attention and feed-forward networks.

For the Summarization-Specific Layer added on top of BERT, it might involve linear transformations, activation functions, or additional attention mechanisms to tailor BERT outputs for summarization defined in Equation (7):

$$Summarization_{layer}\big(BERT_{encoder}(input_{ids})\big) = Decoder\big(BERT_{encoder}(input_{ids})\big) \quad (7)$$

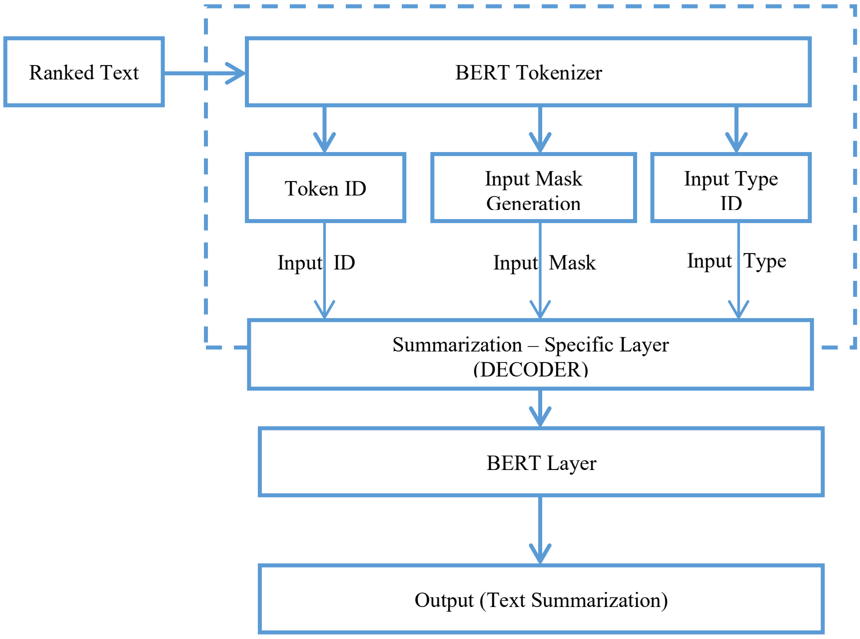

Figure 2. Fine-tuned BERT model

Where: $Summarization_{layer}(.)$ Represents the summarization-specific operations added on top of BERT. $Decoder(.)$ Includes linear transformations, activation functions, or other operations tailored for summarization. During fine-tuning on a summarization-specific dataset, the model optimizes its parameters to minimize a loss function defined in Equation (8);

$$Loss = Loss\left(Summarization_{layer}\left(BERT_{encoder}(input_{ids})\right), target_{summary}\right) \quad (8)$$

Where: $Loss(.)$ Represents a suitable loss function measuring the difference between the generated summary and the target summary. $target_{summary}$ is the ground truth summary from the dataset. During inference on new text, the model makes a summary based on the learned representations as shown in Equation (9):

$$Generated_{summary} = Summarization_{layer}\left(BERT_{encoder}(input_{ids})\right) \quad (9)$$

Where: $Generated_{summary}$ represents the summary generated by the model.

Pseudocode for text summarization using optimized BERT with Summarization-Specific Layer (Decoder)

 # Step 1: Initialization and Setup

tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')

bert_model = BertModel.from_pretrained('bert-base-uncased')

class SummarizationLayer(nn.Module):

def __init__ (self, config):

super (SummarizationLayer, self). __init__ ()

def forward (self, input_ids, attention_mask):

 # Step 2: Data Preprocessing

```
# Tokenize input documents for summarization

# Step 3: Model Definition

summarization_model = SummarizationLayer(config=bert_model.config)

# Step 4: Fine-Tuning

optimizer = AdamW (summarization_model. parameters (), lr=1e-5)

for epoch in range(num_epochs):

  for batch in summarization_data_loader:

      input_ids = batch['input_ids']

      attention_mask = batch['attention_mask']

      target_summary = batch['target_summary']

summary_representation = summarization_model (input_ids, attention_mask)

      loss = calculate_loss (summary_representation, target_summary)

      optimizer. zero_grad ()

      loss.backward()

      optimizer.step()

# Step 5: Inference (Summarization)

def generate_summary(input_text):

  input_ids = tokenizer.encode(input_text, return_tensors='pt')

  summary_representation = summarization_model(input_ids)

  return summary_representation
```

From the above pseudocode, Load the pre-trained BERT tokenizer and model using the $transformers$ library. Define a custom module ($SummarizationLayer$) representing the summarization-specific layers to be added on top of BERT. Define the architecture for the summarization model by combining the BERT encoder with the Summarization-Specific Layer. Initialize an optimizer (AdamW) to update the parameters of the summarization model during training. $generate\_summary$ function tokenizes the input text, passes it through the summarization model, and returns the summary representation.

RESULTS AND DISCUSSIONS

Experimental Analysis

Software and Hardware Configuration

The proposed PMI-driven BERT summarization model was implemented using Python programming language. The deep learning framework PyTorch was utilized for model development and fine-tuning, along with the Hugging Face Transformers library for BERT implementation. Tokenization was performed using the BERT-base-uncased tokenizer. Experiments were conducted on a system with an

Intel Core i7 processor, 16 GB RAM, and a CPU-based execution environment running Windows OS. ROUGE evaluation was carried out using the official ROUGE evaluation toolkit.

Dataset

The CNN / Daily Mail Dataset is a comprehensive collection of English-language news items, with a total of slightly more than 300,000 distinct pieces of journalistic writing produced by reporters affiliated with CNN and the Daily Mail. The existing iteration of the software encompasses both extractive and abstractive summarization techniques. However, it is important to note that the first version was primarily designed for machine reading and comprehension, as well as abstractive question answering. The representation of other types of English in the data remains uncertain.

Data-splits

The dataset provided by CNN/Daily Mail consists of three distinct partitions, namely the training set, the validation set, and the test set. The statistical data pertaining to Version 3.0.0 of the dataset is shown in table 1.

Table 1. Data-splits

Dataset Split	Number of Instances in Split
Train	286,116
Validation	12,468
Test	12,480

Text summaries are generally evaluated using metrics. These are metrics that experts evaluate when creating summaries and they are crucial since professionals will utilise the created tool. Manually evaluating summaries does not scale effectively since it would take enormous amounts of time and effort to review the hundreds, if not thousands, of summaries that exist. As a result, it is critical to supplement human review with qualitative evaluation methodologies and criteria for automatically evaluating summaries.

Data-instances

Each instance in the dataset consists of three strings: one for the article, one for the highlights, and one for the ID as shown in figure 3. Additional cases may be examined to get further insight. Fine-Tuned BERT Model and Summarized Text from Source Data is shown in figure 4 and 5.

```
{'id': '0054d6d30dbcad772e20b22771153a2a9cbeaf62',
 'article': '(CNN) -- An American woman died aboard a cruise ship that docked at Rio
de Janeiro on Tuesday, the same ship on which 86 passengers previously fell ill,
according to the state-run Brazilian news agency, Agencia Brasil. The American
tourist died aboard the MS Veendam, owned by cruise operator Holland America.
Federal Police told Agencia Brasil that forensic doctors were investigating her
death. The ship's doctors told police that the woman was elderly and suffered from
diabetes and hypertension, according the agency. The other passengers came down with
diarrhea prior to her death during an earlier part of the trip, the ship's doctors
said. The Veendam left New York 36 days ago for a South America tour.'
```

Figure 3. Text from CNN/daily mail dataset

```
bert_model = Summarizer()
bert_summary = ''.join(bert_model(body, min_length=60))
print(bert_summary)
```

Figure 4. Fine-tuned BERT model for summarization

```
'highlights': 'The elderly woman suffered from diabetes and hypertension, ship's
doctors say .\nPreviously, 86 passengers had fallen ill on the ship, Agencia Brasil
says .'}
```

Figure 5. Summarized text from source data

The following data shows the average token count for both the articles and the highlights as given in table 2.

Table 2. Mean token count

Feature	Mean Token Count
Article	785
Highlights	57

Performance Evaluation

Precision, Recall and F-Score

Precision, recall, and F-Score are regularly used metrics in text summarization assessment to compare the performance of a summarising system to a reference summary.

Precision is the percentage of accurately identified information in the produced summary that corresponds to the information in the reference summary.

The percentage of information in the reference summary that is properly recognised in the produced summary is measured by recall.

The F-Score (F1-Score) is the harmonic mean of accuracy and recall, resulting in a single statistic that balances precision and recall.

The evaluation metrics Precision, Recall, and F-Score are defined using separate mathematical expressions. Equation (10) represents Precision and quantifies the proportion of correctly predicted positive instances. Equation (11) defines Recall, measuring the ability of the model to identify all relevant positive instances. Equation (12) describes the F-Score, which provides a harmonic balance between Precision and Recall. Each equation is numbered and cited individually along with its corresponding description to ensure clarity and methodological rigor.

$$Precison = \frac{Number\ of\ correct\ items\ in\ the\ generated\ summary}{Total\ number\ of\ items\ in\ the\ generated\ summary} \quad (10)$$

$$Recall = \frac{Number\ of\ correct\ items\ in\ the\ generated\ summary}{Total\ number\ of\ items\ in\ the\ reference\ summary} \quad (11)$$

$$F - Score = \frac{2\ \times Precision\ \times Recall}{Precision + Recall} \quad (12)$$

Where:

"Number of correct items" refers to the relevant information or content that matches between the generated summary and the reference summary.

"Total number of items" signifies the total content or information in either the generated summary or the reference summary, depending on whether it's precision or recall.

Rouge

ROUGE (Recall-Oriented Understudy for Gisting Evaluation), encompasses a collection of metrics used for the automated assessment of the efficacy of a produced summary. This evaluation is conducted by comparing the created summary to one or more reference summaries. The evaluation metric quantifies the degree of similarity between the produced summary and the reference summaries by considering factors such as n-gram overlap, word sequences, and other similarity metrics. The ROUGE measures that are often used include:

ROUGE-N (ROUGE-Ngram):

It processes the overlap of n-grams (sequences of n consecutive words) between the generated summary and reference summaries. ROUGE-1 (unigrams), ROUGE-2 (bigrams), ROUGE-3 (trigrams), etc. For ROUGE-1 (unigram overlap), the evaluation metrics are computed as follows. Equation (13) defines ROUGE-1 Precision, which measures the proportion of overlapping unigrams relative to the total number of unigrams in the generated summary. Equation (14) represents ROUGE-1 Recall, which quantifies the proportion of overlapping unigrams with respect to the total number of unigrams in the reference summary. Equation (15) defines the ROUGE-1 F1-Score, which computes the harmonic mean of ROUGE-1 Precision and ROUGE-1 Recall to provide a balanced performance measure.

- *Rouge-1(unigram overlap):*

$$Precision: ROUGE-1_{Precision} = \frac{Count\ of\ overlapping\ unigrams\ in\ generated\ summary\ and\ reference}{Total\ unigrams\ in\ generated\ summary} \tag{13}$$

$$Recall: ROUGE-1_{recall} = \frac{Count\ of\ overlapping\ unigrams\ in\ generated\ summary\ and\ reference}{Total\ unigrams\ in\ reference\ summary} \tag{14}$$

$$F1-Score: ROUGE-1_{F1} = \frac{2\ \times ROUGE-1_{Precision} \times ROUGE-1_{Recall}}{ROUGE-1_{Precision} +\ ROUGE-1_{Recall}} \tag{15}$$

- ROUGE-2 (bigram overlap), ROUGE-3 (trigram overlap), etc., follow similar precision, recall, and F1-score formulas but consider different n-gram sizes.

ROUGE-L (ROUGE-Longest Common Subsequence):

The metric quantifies the length of the longest collective subsequence among the summary created by the system and the reference summaries. Additionally, it is important to consider the structure of sentences and the arrangement of words.

ROUGE-W (ROUGE-Weighted n-gram):

Similar to ROUGE-N, but the algorithm gives more weights to n-grams. at the beginning of the sentences. It helps in capturing the importance of initial words.

ROUGE-S (ROUGE-Skip-bigram):

It processes the overlap of skip-bigrams (n-grams with specified gaps between words) between the generated summary and reference summaries. Also, captures the semantic meaning even if the words are not consecutive.

ROUGE-SU (ROUGE-Skip-bigram with Unigram):

This approach utilises a combination of skip-bigrams and unigrams to assess the efficacy of the produced summary.

The quality of the created summary is assessed using ROUGE metrics, which calculate precision, recall, and F1-score. These measures refer to how similar the generated summary is to the reference summaries by counting the similarity between the number of n-grams or sub-sequences in table 3. A greater ROUGE score indicates a stronger concurrence between the produced summary and the reference summaries as shown in Figure 6.

Table 3. ROUGE metrics

Length	**10%**		
Metric	Recall	Precision	F-Score
Rouge-1	1	0.82	0.86
Rouge-2	0.97	0.82	0.84
Rouge-L	1	0.74	0.86

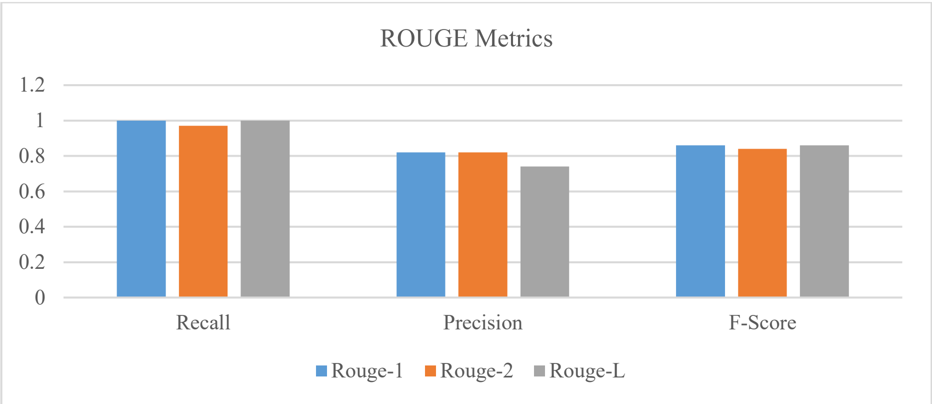

Figure 6. Performance comparison of ROUGE metrics across different summarization models

Table 4. Rouge score of difference methods

Model	**R-1**	**R-2**	**R-L**	**R-Average**
seq-to-seq [13]	44.76	25.26	42.18	37.38
Seq2sick [23]	42.86	23.46	38.48	35.63
Prefix-tuning [17]	45.72	26.43	43.93	38.69
Proposed	46.9	27.61	45.68	40.62

Table 4 shows the comparative outcomes of the planned work using baseline approaches.

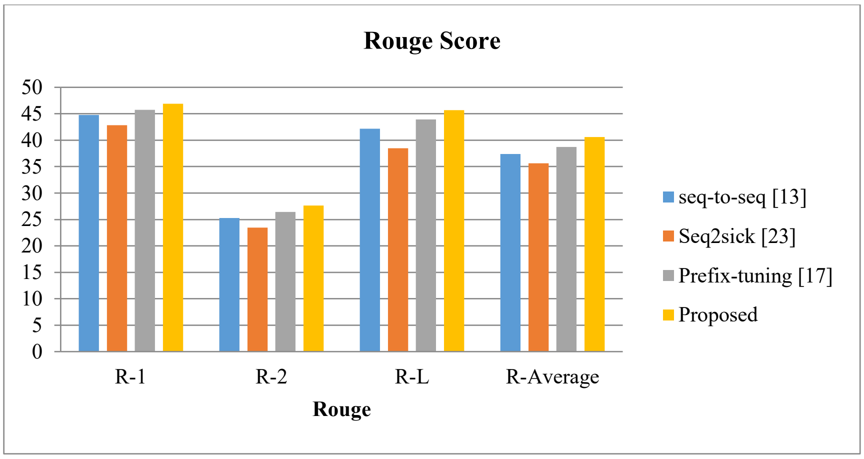

Figure 7. Rough score

Figure 7 shows a graphical depiction of table 4. According to the findings, the suggested model beat the current baseline methods.

Ablation Study

An ablation study was conducted to analyze the impact of PMI-based pre-processing on the summarization performance. The proposed model was evaluated under two configurations: (i) Conventional BERT without PMI pre-processing and (ii) PMI-driven BERT summarization framework. Results indicate that removing PMI leads to a noticeable reduction in ROUGE scores, particularly in ROUGE-2 and ROUGE-L metrics. This confirms that PMI-based term ranking significantly contributes to improved semantic relevance and contextual coherence in the generated summaries.

CONCLUSION

The achievement of a PMI-driven solution containing the mighty contextual understanding of traditional BERT is a major step toward improving the text summarization strategies. This method will improve the summarization process and help improve the quality of generated summaries by leveraging the statistical richness of PMI scoring preprocessing model and the contextual depth provided by BERT. This strategy will narrow down the summarization process through the additions of Decoder of summarization-specific layer that is overlaid on top of BERT Model by utilizing statistical associations of ranking terms based on PMI and the contextual interpretation of the language encoded with the BERT. The evaluation metrics, including ROUGE scores, reveal the fact that the PMI method, combined with Fine-Tuned BERT, proves to have a better result in the form of summary generation which would reflect the necessary and important information of the original documents. This paper is restricted to the application of CPU usage as the predictor variable. The research could be improved as future studies by adding other variables like memory and hard disk-usage. The fact that these variables have been included can help in the creation of more multivariate and comprehensive predictive model. In addition, it would prove beneficial to evaluate the predictive power of the models using additional datasets. The proposed model achieved an average ROUGE improvement of approximately 2–3% over existing baseline methods, demonstrating statistically meaningful enhancement in summarization quality. The integration of PMI significantly improves semantic prioritization, while BERT ensures contextual coherence. These improvements validate the effectiveness and practical applicability of the proposed framework.

REFERENCES

[1] Belwal RC, Gupta A. Automatic text summarization techniques: categorization and contemporary challenges. Information Processing and Management. 2025;62(2):103612.

[2] Aswani S, Choudhary K, Shetty S, Nur N. Automatic text summarization of scientific articles using transformers—A brief review. Journal of Autonomous Intelligence. 2024;7(5). https://doi.org/10.32629/jai.v7i5.1331

[3] Wibawa AP, Kurniawan F. A survey of text summarization: Techniques, evaluation and challenges. Natural Language Processing Journal. 2024 Jun 1;7:100070. https://doi.org/10.1016/j.nlp.2024.100070

[4] Zhang Y, Jin H, Meng D, Wang J, Tan J. A comprehensive survey on automatic text summarization with exploration of LLM-based methods. Neurocomputing. 2025 Nov 3:131928. https://doi.org/10.1016/j.neucom.2025.131928

[5] Liu W, Sun Y, Yu B, Wang H, Peng Q, Hou M, Guo H, Wang H, Liu C. Automatic text summarization method based on improved Text Rank algorithm and K-means clustering. Knowledge-Based Systems. 2024 Mar 5;287:111447. https://doi.org/10.1016/j.knosys.2024.111447

[6] Motghare M, Agarwal M, Agrawal A. NewsSumm: The World's Largest Human-Annotated Multi-Document News Summarization Dataset for Indian English. Computers. 2025;14(12):508. https://doi.org/10.3390/computers14120508

[7] Ghanem FA, Padma MC, Abdulwahab HM, Alkhatib R. Deep Learning-Based Short Text Summarization: An Integrated BERT and Transformer Encoder–Decoder Approach. Computation. 2025 Apr 12;13(4):96. https://doi.org/10.3390/computation13040096

[8] Kaushal A. Charting the growth of text summarization: a bibliometric analysis. Applied Sciences. 2024;14(23):11462.

[9] Chen X, Smith J. Advances in biomedical text summarization: methods and evaluation strategies. Journal of Biomedical Informatics. 2024;149:104302.

[10] Rani Krishna KM, Singh R. Deep learning approaches for automatic text summarization: a comparative analysis. Scientific Reports. 2025;15:20224.
[11] Durak HY, Egin F, Onan A. Multi-agent deep learning framework for enhanced text summarization. Expert Systems with Applications. 2025;240:122324.
[12] Luo M, Zhao L. Automatic text summarization techniques: recent advances and open challenges. Information Retrieval Journal. 2024;27(4):423–445.
[13] Hu X, Li Y, Zhao J. Semantic feature learning for document-level text summarization. Journal of Intelligent Information Systems. 2025;64(2):389–408.
[14] Wang Q, Lee S. Graph neural network architectures for document summarization. IEEE Transactions on Neural Networks and Learning Systems. 2024;35(9):11245–11257.
[15] Ruiz C, Martinez L. Cross-lingual text summarization using attention-based transformer models. Neural Computing and Applications. 2025;37(6):4571–4586.
[16] Patel D, Arora S. Hybrid extractive–abstractive text summarization using deep neural networks. Applied Soft Computing. 2024;143:110467.
[17] Chen L, Kumar A. Reinforcement learning-based abstractive text summarization. Neurocomputing. 2025;559:126804.
[18] Subalakshmi N, Babubalaji R. Improving the Accuracy of Sarcasm Detection in Text Data Using a Smooth Support Vector Classification Model with Word-Emoji Embedding for News and Indian Indigenous Languages. In2024 International Conference on System, Computation, Automation and Networking (ICSCAN) 2024 Dec 27 (pp. 1-6). IEEE. https://doi.org/10.1109/ICSCAN62807.2024.10894530
[19] Zhao T, Tang Y. Context-aware text summarization for legal documents. Artificial Intelligence Review. 2025;58(3):1751–1773.
[20] Lopez G, Silva R. Transformer-based multilingual text summarization. Natural Language Engineering. 2024;30(5):843–862.
[21] Park JH, Cho M. Scalable neural text summarization in big data environments. Big Data Research. 2025;35:100392.
[22] Ahmed S, Rahman N. Domain adaptation techniques for neural text summarization. Pattern Recognition Letters. 2024;176:77–84.
[23] Ozdemir A, Yilmaz T. Integrating semantic role labeling with transformer models for text summarization. Computational Linguistics. 2025;51(1):129–154.
[24] Barrios J, Fernandez M. Beyond ROUGE: evaluation metrics for text summarization. Information Processing & Management. 2024;61(6):103271.
[25] Nair R, Dev P. Recent trends in text summarization evaluation and benchmarks. Journal of Artificial Intelligence Research. 2025;72:1159–1186.
[26] Jeyakarthic M, Senthilkumar J. Optimal bidirectional long short-term memory-based sentiment analysis with sarcasm detection and classification on twitter data. In2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon) 2022 Oct 16 (pp. 1-6). IEEE. https://doi.org/10.1109/MysuruCon55714.2022.9972540
[27] Jeyakarthic M, Leoraj A. A novel social media-based adaptable approach for sentiment analysis data. In2023 Second International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT) 2023 Apr 5 (pp. 1-6). IEEE.
[28] Leoraj A, Jeyakarthic M. Spotted Hyena Optimization with Deep Learning-Based Automatic Text Document Summarization Model. International Journal of Electrical and Electronics Engineering. 2023;10(5):153-64. https://doi.org/10.14445/23488379/IJEEE-V10I5P114