

A Comparative Study of Semantic and Syntactic Approaches for Abstractive Text Summarization using BART and T5 Transformers

Manisha Sachin Dabade*

Department of Computer Engineering, BRAC's Vishwakarma Institute of Information Technology, Savitribai Phule Pune University, Pune, India

Department of Computer Science and Engineering, Walchand College of Engineering, Sangli, Maharashtra, India.

E-mail: manisha.221p0072@viit.ac.in

ORCID iD: <https://orcid.org/0009-0000-9038-317X>

*Corresponding author

Vishal Meshram

Department of Computer Engineering, of S&T, Vishwakarma University, Pune, India

E-mail: vishal.meshram@vupune.ac.in

ORCID iD: <https://orcid.org/0000-0001-9950-584X>

Received: April 10, 2026; Revised: June 22, 2026; Accepted: August 1, 2026; Published: October 8, 2026

Abstract: Automatic text summarization plays an important role in transforming lengthy documents into concise and informative representations. The abstractive summarization aims to generate sentences that capture the semantic meaning of the source text. This study proposed semantic and syntactic augmentation in transformer-based abstractive summarization using two encoder-decoder models such as a BART-based model enhanced with SpaCy Part-of-Speech analysis and RoBERTa-based semantic embeddings, and a prompt-guided T5 model, in which task-specific textual prompts were appended to the input sequence to guide the summarization process during fine-tuning and inference. Both models were trained and evaluated on the BBC News dataset using ROUGE parameters. The final results show that the T5 model obtained ROUGE-1 of 45.61, ROUGE-2 of 30.12, and ROUGE-L of 45.12. The BART model obtained ROUGE-1 of 44.28, ROUGE-2 of 28.70, and ROUGE-L of 44.72. The results indicate that the prompt-based T5 model combined with linguistic feature augmentation can improve the quality of abstractive summaries. However, as the reference summaries in news datasets contain extractive characteristics, ROUGE-based evaluation reflects lexical overlap in addition to abstractive generation. The final findings show the improved abstractive summarization performance within the constraints of the dataset and evaluation protocol.

Index Terms: Natural Language Processing, Text Summarization, Transformer Models, BART, T5, Semantic Features, Syntactic Features.

1. Introduction

In the age of information overload, the ability to automatically condense large volumes of textual data into concise and consistent summaries becomes increasingly essential [1]. Text summarization serves as a critical tool to enable rapid comprehension and informed decision-making from news media and social networks to legal and medical records [2]. Natural Language Processing (NLP) has significantly advanced the area through the development of deep learning (DL) models that understand context, syntax, and semantics, allowing machines to generate relevant summaries with minimal human intervention [3, 4]. Text summarization is a process of reducing a document or a set of documents into a shorter version to preserve important information [5, 6]. There are two primary types of summarization techniques [7]

Extractive Summarization aims to identify and extract key sentences or phrases directly from the original text [8]. It retains the exact wording and structure of the source, depending on statistical or linguistic features such as term frequency, sentence position, or graph-based rankings. For example, the TextRank algorithm selects the most "important" sentences based on sentence connectivity [9].

Abstractive Summarization involves paraphrasing the original content using new sentences that convey the same meaning [10]. This approach is more human-like and requires a deeper understanding of the text that involves neural

network models [11]. The abstractive model generates "The prime minister announced new reforms today" based on a paragraph describing a press conference.

The transformer models achieved significant outcomes in abstractive summarization; the integration of syntactic and semantic information helps generate summaries more effectively [12]. This research addresses this gap by proposing and comparing two models that fuse linguistic features into the summarization strategy and another that uses task reformulation through prompting. The novelty lies in the dual evaluation of syntax-semantic enhancement versus prompt-based modeling under a consistent experimental approach [13].

This study investigates the semantic and syntactic augmentation to improve transformer-based summarization quality. The work focused on comparing two transformer models, BART and T5 using linguistic enhancement and prompt-based reformulation on the BBC News dataset. The objective is to measure how the strategies influence summary quality, model behavior, and performance as measured by standard summarization parameters. The core contributions of this paper are summarized as follows:

- To propose a syntax-semantic enriched BART model, incorporating SpaCy-based POS tagging and RoBERTa-based semantic embeddings to augment the summarization.
- To implement a prompt-based T5 model for transformer-based text summarization as a text-to-text task using natural language prompts.
- To present a comparative analysis of both models in terms of training dynamics, summary quality, and evaluation performance.
- To analyze the role of semantic and syntactic cues to improve summary generation quality and to discuss the limitations of ROUGE-based evaluation for distinguishing abstraction from copying

The rest of the paper is organized as follows:

Section 2 reviews related work on extractive, abstractive, and linguistically informed summarization methods. Section 3 describes the proposed methodology in detail. Section 4 presents the experimental results of proposed models. Section 5 discusses the comparative performance and qualitative results of models. Section 6 concludes the paper and discusses the directions for future research.

2. Related Works

The field of automatic text summarization has shown significant improvement over the past two decades due to the advancement of DL and transformer-based models. The existing summarization techniques are categorized into extractive and abstractive methods. Each type has unique characteristics, capabilities, and limitations. The recent works explored the integration of linguistic cues to enhance the quality and consistency of generated summaries.

2.1. Extractive Summarization Techniques

Extractive summarization refers to selecting the subset of sentences or phrases directly from the source document without altering the original text. The methods are computationally efficient and maintain grammatical correctness but lack cohesion and contextual depth.

In previous decades, investigators studied the extractive text summarization from various perspectives and using various techniques. Li et al. suggested DL data-driven models to generate extractive summaries. A paraphrase technique was used to determine which words had to be featured in the summary. In the model, tightly coupled layers of cells and a feature mapping were produced by a convolutional layer. For every expression, two results were generated for each category, and evaluation parameters were employed for assessing rather than ROUGE because summary creation is a binary classification problem. The suggested model obtained an accuracy of 90.00%. This resulted from the dataset being derived from human summaries [14]. Another approach, proposed by Kumar et al. [15], is one in which textual expressions serve as nodes, and the significance of the edge connecting sentences indicates the relationship between them. A customized inverted phrase frequency-cosine similarity was created to give different weights to different phrases in the written content, as opposed to regular cosine similarity, which analyzes words similarly. There was little division of the visualization into other categories. It is predicated on the idea that the phrases within a cluster are comparable to each other. The effectiveness of the suggested summary technique was shown by the outcomes of the assessment. Graph-based summary extraction was studied by Jang and Kang [16]. The method took into account how related the nodes located on the graph's edges are to one another. Additionally, weights were assigned according to how related the issue was. The similarity across nodes was also determined using semantic measures. The evaluation parameters of the suggested approach were, respectively, 0.154, 0.229, and 0.445.

2.2. Abstractive Summarization Techniques

Abstractive summarization generates new sentences that paraphrase the source content, often requiring deeper semantic understanding and language generation capabilities.

Sequence-to-Sequence (Seq2Seq) Models: Introduced by Tian Shi et al. (2021), use an encoder-decoder architecture with RNNs and LSTMs to translate input text into summaries. The suggested model struggled with long-

range dependencies and factual consistency [17].

Pointer-Generator Networks (Li Yuanyuan et al., 2023): Combine abstractive generation with extractive copying mechanisms to handle out-of-vocabulary words and improve factual accuracy [18].

In contrast to extractive summarization, abstractive text summarization techniques have been developed. The study in [19] is one of the investigations. The problem of producing inaccurate data about the source material in abstractive summarization was tackled by the researchers. A set of two actual improved techniques known as SpanFact was employed to address this problem. According to the CNN dataset, the ROUGE score was 19.27 for R-2, 41.75 for R-1, and 38.81 for R-L. An abstractive summary that is topic-guided was proposed by Mutlu et al. [20]. Their strategy guarantees a certain amount of reliance on the text's subject. As part of their seq2seq transformer modeling, researchers incorporated content modeling.

Results from evaluating their suggested approach on the CNN dataset were 41.33 for R-L, 21.19 for R-2, and 44.38 for R-1. A technique for abstractive text summarization using GAN was presented by Patel et al. [21]. Both a discriminator and a summary generator are part of their intended model. The summary is created by the generator, and the discriminator attempts to distinguish between a human-generated and a machine-generated summary. According to the results, the R-1, R-2, and R-L scores were 37.87, 15.71, and 39.20, respectively. The use of LSTM-CNN for abstractive text summarization was suggested by Chan and King in [22]. Terms were initially taken from their system. Following the phrase extraction procedure, an LSTM-CNN summary was produced. The testing measures, ROUGE-1 and ROUGE-2, yielded results of 34.9 and 17.8, respectively. A PAN model for abstractive text summarization was proposed by Espino et al. [23]. A pre-trained layer of word embedding was proposed in order to solve the problem of the network producing words that were not part of the lexicon. The R-1, R-2, and R-L scores were 39.06, 17.05, and 35.85, respectively.

The transformer-based models such as BART and T5 were used for abstractive summarization. The behavior of a summarization depends not only on model architecture but also on the nature of the training targets and evaluation protocol. When reference summaries were strongly extractive directly from source-leading sentences, ROUGE-based gains partly show lexical copying rather than genuine periphrastic abstraction. The present work focused on a comparative analysis of semantic and syntactic enhancement strategies in transformer-based summarization to explicitly acknowledge the distinction between extractive overlap and true abstraction.

3. Proposed Methodology

This section describes the design and implementation of two deep learning models for abstractive text summarization:

- Model 1: A linguistically enriched BART model used syntactic and semantic features.
- Model 2: The T5 model is fine-tuned using a prompt-guided training strategy. The T5 model formulates all tasks as text-to-text generation; explicit task prompts are appended to the input sequence to guide the summarization.

Both models were evaluated using the same dataset and preprocessing strategy for a fair and consistent comparison. The POS embeddings and RoBERTa contextual embeddings were concatenated with BART token embeddings and projected through a linear transformation before being fed into the BART encoder.

3.1. BBC News Dataset

It is a collection of 2,225 documents extracted from the BBC news website, representing stories published between 2004 and 2005. These documents are categorized into five topical areas: business, entertainment, politics, sport, and tech [24]. The BBC News Kaggle Archive dataset does not provide predefined summaries such as headlines or bullet-point abstracts. Therefore, reference summaries used in this study were constructed during preprocessing to create supervised training targets. The summaries were generated using sentence-level extraction methods that select informative sentences from the source document rather than relying solely on the headline or the first sentence. This approach ensures the summarization task remains non-trivial and requires the model to learn meaningful content. The source articles contain approximately 380 tokens on average, and the reference summaries contain approximately 40 tokens, indicating an average compression ratio of approximately 10:1. This shows that the model significantly condenses the source document to retain the essential information and make the task more challenging than simple headline generation.

3.2. Data Preprocessing

In this study, the target summaries are constructed using extractive-style references derived from the source articles through sentence-level selection. The BART and T5 are generative encoder-decoder models. The learning objective is supervised using summaries with high lexical overlap to the input text. This experimental design enables controlled comparative evaluation of the models but does not, by itself, establish fully abstractive paraphrase generation. The datasets contain multiple articles covering the same event; duplicate documents were examined during preprocessing.

Articles with highly similar text were removed using text similarity filtering to reduce overlap between training and test sets. This step helps prevent the model from memorizing the specific articles and shows the model's generalization capability. In this step, the dataset is standardized and prepared for training the proposed models. All text is converted to lowercase and cleaned using normalized whitespace. The input documents were truncated to a maximum length of 512 tokens because summaries are limited to 128 tokens to maintain consistency. Finally, tokenization is performed using model-specific tokenizers for Model 1 *BARTTokenizer* and *T5Tokenizer* for Model 2 ensuring compatibility with each model. Let the raw input document be denoted as:

$$D = [w_1, w_2, \dots, w_n] \quad (1)$$

where w_i represents the i^{th} word in the original text and n is the total number of words.

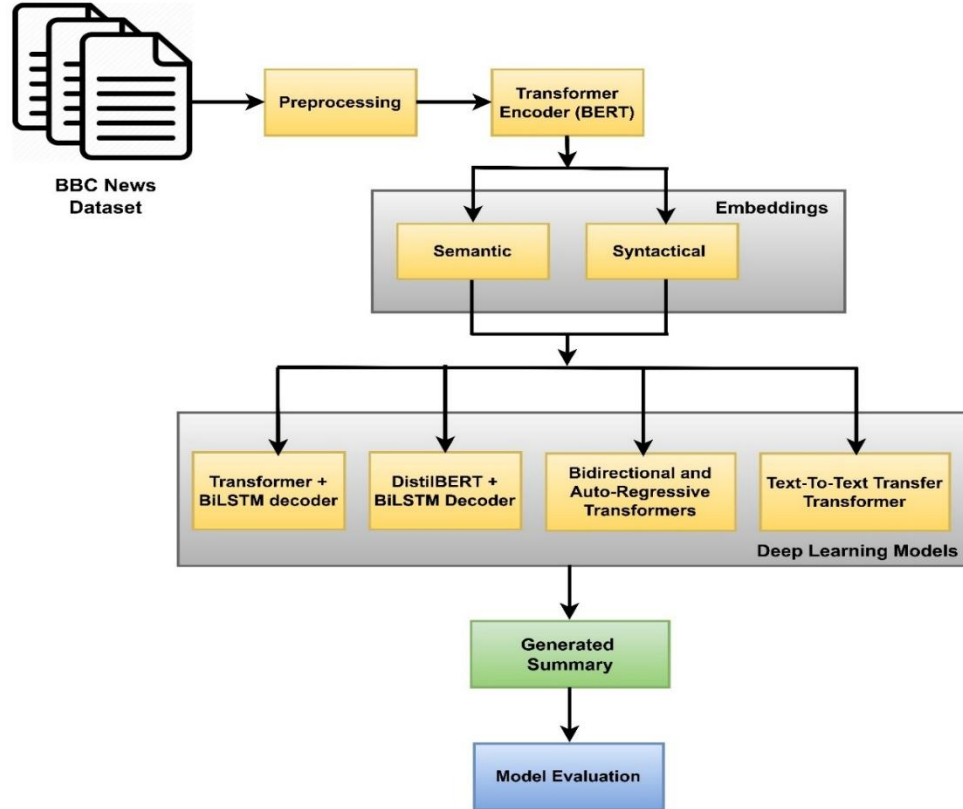


Fig. 1. Architecture of the proposed models for semantic and syntactical based text summarization

Step 1: Lowercase and Normalization: Apply a normalization function f_{norm} , which performs lowercasing and removes redundant whitespaces:

$$D' = f_{norm}(D) = [lower(w_1), lower(w_2), \dots, lower(w_n)] \quad (2)$$

Step 2: Tokenization: Use a model-specific tokenizer $T_m \in \{BARTTokenizer, T5Tokenizer\}$ to convert the normalized sequence into tokens:

$$X = T_m(D') = [x_1, x_2, \dots, x_T], \text{ where } T \leq 512 \quad (3)$$

Step 3: Summary Tokenization: Similarly, for the corresponding reference summary S , we have:

$$Y = T_m(S) = [y_1, y_2, \dots, y_L], \text{ where } L \leq 128 \quad (4)$$

The raw document D is first normalized using a function f_{norm} that applies lowercasing and whitespace cleaning. The cleaned document D' is then tokenized using a model-specific tokenizer T_m , which converts it into a fixed-length token sequence X , truncated to a maximum of 512 tokens. The target summary S is also tokenized into Y , truncated to 128 tokens, to serve as the decoder target during training.

3.3. Building the Model

Bidirectional and Auto-Regressive Transformers: This section presents the linguistically enriched approach to abstractive summarization that combined the syntactic structure and semantic understanding into a generative model. The proposed model consists of three core components: syntactic feature extraction using SpaCy, semantic feature extraction using RoBERTa, and the summarization process using BART. These components were combined within a custom PyTorch-based strategy to augment the input representation for effective training and evaluation.

Integration of Semantic and Syntactic Features into BART: The proposed model incorporated the semantic embeddings derived from RoBERTa and syntactic information obtained through SpaCy Part-of-Speech tagging into the BART encoder to improve the contextual representation of input text. This integration enhances the token representations before they are processed by the transformer encoder. For each input token t_i three types of representations were extracted:

BART token embedding: $E_{bart}(t_i)$, obtained from the pretrained BART tokenizer and embedding layer.

RoBERTa contextual embedding $E_{roberta}(t_i)$, extracted from the last hidden layer of the pretrained RoBERTa encoder.

POS tag embedding $E_{pos}(t_i)$, generated by mapping each POS tag to a learnable embedding vector. The representations were combined through feature concatenation and produced a unified token representation:

$$E_{combined}(t_i) = [E_{bart}(t_i) \parallel E_{roberta}(t_i) \parallel E_{pos}(t_i)] \quad (5)$$

where $\parallel$ is the vector concatenation. To maintain compatibility with the BART encoder dimension, the concatenated representation is projected using a linear transformation:

$$E_{final}(t_i) = W_p \cdot E_{combined}(t_i) + b_p \quad (6)$$

where W_p and b_p are the learnable parameters. The resulting vector $E_{final}(t_i)$ is then passed as input to the BART encoder layers. This strategy allows the model to combine the additional semantic and syntactic cues to preserve the standard transformer model.

Syntactic Feature Extraction using SpaCy: In this study, SpaCy Part-of-Speech tagging is used as a lightweight syntactic filter to highlight tokens that capture the essential semantic information, such as nouns, verbs, and proper nouns. This step does not modify the internal architecture of the transformer model that helps to identify salient tokens before training. The POS tagging serves as an auxiliary linguistic cue that complements the model's implicit syntactic learning rather than replacing it. Let a document be represented as a sequence of tokens:

$$X = [x_1, x_2, \dots, x_T] \quad (7)$$

where x_i is the i th token and T is the total number of tokens. The syntactic feature extractor maps this sequence to a POS tag sequence:

$$P = POS(X) = [p_1, p_2, \dots, p_T] \quad (8)$$

where $p_i \in P$ and P is the POS tag set ($\{NOUN, VERB, ADJ, \dots\}$).

The POS tagging component is used solely during preprocessing to guide token selection and does not alter the structure of the transformer encoder. BART's attention mechanisms capture syntactic dependencies, while explicit POS information helps to emphasize structurally important words that correspond to entities, actions, and events in news articles. Each POS tag produced by SpaCy is mapped to a trainable embedding vector. The embeddings indicate the lightweight syntactic signals that show the grammatical roles such as nouns, verbs, adjectives, and proper nouns. This strategy aims to enhance feature representation rather than introduce manual feature engineering into the neural model.

Semantic Embeddings using RoBERTa: To enhance contextual understanding beyond surface-level syntax, we extract semantic representations using the RoBERTa-base transformer model. Each input document is passed through RoBERTa to generate contextual token-level embeddings. Given the input token sequence X , RoBERTa computes a contextual representation:

$$H = RoBERTa(X) = [h_1, h_2, \dots, h_T] \quad (9)$$

where $h_i \in R^d$ and $d = 768$ for roberta-base. This representation captures rich semantic dependencies and relationships between tokens in the context of the entire document. For practical efficiency, the embeddings are computed and stored in advance using:

$$H_{semantic} = f_{sem}(X) \quad (10)$$

where f_{sem} denotes the frozen RoBERTa encoder. The RoBERTa encoder is used as an external semantic representation module to generate contextual embeddings for each token. The embeddings capture rich semantic relationships learned during large-scale pretraining. In the proposed model, RoBERTa embeddings were extracted for summarization and combined with BART token embeddings to enhance contextual understanding of the source document.

Text Generation using BART: The BART (Bidirectional and Auto-Regressive Transformers) model is used as the summarizer in the proposed model. BART is a transformer-based sequence-to-sequence model that combines a bidirectional encoder (like BERT) and a left-to-right autoregressive decoder. Let X be the input document and $Y = [y_1, y_2, \dots, y_L]$ be the corresponding target summary. The BART encoder computes:

$$Z = \text{EncoderBART}(X) \quad (11)$$

The decoder generates each output token autoregressively:

$$P(y_t | y_{<t}, Z) = \text{DecoderBART}(y_{<t}, Z) \quad (12)$$

The objective is to maximize the conditional likelihood of the reference summary given the document:

$$L_{BART} = -\sum_{t=1}^L \log P(y_t | y_{<t}, Z) \quad (13)$$

The model is fine-tuned on the BBC News dataset using this loss function and optimized via the Adam optimizer.

Custom Dataset Implementation: A custom PyTorch dataset class was designed to efficiently manage both textual and semantic representations during training. Each instance in the dataset includes:

Tokenized Input: The original document tokenized using BartTokenizer.

Target Summary: The corresponding summary (first two sentences) also tokenized using BartTokenizer.

Semantic Embedding: Precomputed RoBERTa embeddings stored as tensors of *shape* (T, d) where $T \leq 512$ and $d = 768$. Formally, for the i th training sample, the dataset returns:

$$D(i) = \{\text{input}_{ids(i)}, \text{attention}_{mask(i)}, \text{target}_{ids(i)}, H_{semantic}(i)\} \quad (14)$$

Model 2: Prompt-based T5: In this model, we leverage the T5 (Text-to-Text Transfer Transformer) architecture for abstractive summarization using a prefix-based instruction format. The T5 model allows summarization to be performed by prepending a natural-language prompt to the input text and generating the corresponding summary.

Prefix-based Input Encoding: The key innovation in T5's formulation lies in its prompt-based input construction. A special prefix keyword instructs the model to perform a specific task. For summarization, the format is:

$$\text{Input T5} = \text{summarize:} + X \quad (15)$$

Let $X = [x_1, x_2, \dots, x_T]$ represent the input document. Then the T5 input is:

$$X' = [\text{summarize:}, x_1, x_2, \dots, x_T] \quad (16)$$

This sequence is tokenized using the T5Tokenizer, which maps tokens to subword indices and generates the corresponding attention masks. The maximum input length is fixed to 512 tokens, and the summary is limited to 128 tokens. The target summary $Y = [y_1, y_2, \dots, y_L]$ is tokenized similarly to serve as the decoder output.

Sequence-to-Sequence Summarization with T5: The T5 model follows the standard encoder-decoder transformer model. The model is pre-trained using a denoising objective on a large corpus and then fine-tuned for downstream tasks like summarization.

Encoder Step: Given the tokenized input X' the encoder computes:

$$Z = \text{EncoderT5}(X') \quad (17)$$

where $Z \in \mathbb{R}^T \times d$ and d is the hidden dimension of the T5 model (768 for t5-base).

Decoder Step: The decoder generates the summary autoregressively, predicting each token conditioned on the previous tokens and the encoder output:

$$P(y_t | y_{<t}, Z) = \text{DecoderT5}(y_{<t}, Z) \quad (18)$$

The objective is to maximize the conditional likelihood of the output summary:

$$LT5 = -t = 1 \sum \log P(y_t | y_{<t}, Z) \quad (19)$$

This is implemented using cross-entropy loss over the vocabulary.

Training and Inference Workflow: The training procedure optimizes the conditional likelihood of the summary tokens using the Adam optimizer. During inference, the model uses beam search decoding (greedy decoding) to generate fluent and concise summaries.

$$L_{train} = -\frac{1}{N} \sum_{i=1}^N \sum_{t=1}^{L_i} \log P(y_t^{(i)} | y_{<t}^{(i)}, X'^{(i)}) \quad (20)$$

where N is the number of training examples, L_i is the length of the i^{th} summary

Algorithm 1: BART + Syntax + Semantics-based Summarization

Input: BBC News dataset with full articles

Output: Abstractive summaries using BART

```

1: Load SpaCy POS tagger and RoBERTa semantic encoder
2: Load BART tokenizer and BART summarization model
3: Initialize empty list: texts, summaries
4: for each file in dataset:
5:   Read full article as text
6:   Generate extractive summary (first 2 sentences)
7:   Append text and summary to respective lists
8: end for
9: Define function Extract_SyntacticFeatures(text):
10:  doc ← SpaCy(text)
11:  pos_tags ← [token.pos_for token in doc]
12:  return pos_tags
13: end function
14: Define function Extract_SemanticEmbedding(text):
15:  tokens ← RoBERTaTokenizer(text)
16:  embedding ← RoBERTaModel(tokens)
17:  return embedding.last_hidden_state
18: end function
19: Define Custom Dataset:
20:  for each (text, summary) pair:
21:    input_ids ← BARTTokenizer(text)
22:    target_ids ← BARTTokenizer(summary)
23:    semantic_embedding ← Extract_SemanticEmbedding(text)
24:    yield input_ids, target_ids, semantic_embedding
25:  end for
26: end Dataset
27: Split dataset into training and validation sets
28: forepoch = 1 to  $N$  do:
29:  for each batch in training_loader do:
```

```
30:   Compute loss using  $BART(input_{ids}, target_{ids})$ 
31:   Backpropagate and update model weights
32: end for
33:   Compute validation loss
34: end for
35: Evaluate model using ROUGE scores
36: Return generated summaries
```

Algorithm 2: Prompt-based T5 Summarization

Input: BBC News dataset with full articles

Output: Abstractive summaries using T5 model

```
1: Load T5 tokenizer and T5 summarization model
2: Initialize empty list: texts, summaries
3: for each file in dataset:
4:   Read full article as text
5:   Generate extractive summary (first 2 sentences)
6:   Append text and summary to respective lists
7: end for
8: Define CustomT5Dataset:
9:   for each (text, summary) pair:
10:     $input_{text} \leftarrow \text{"summarize:"} + text$ 
11:     $input_{ids} \leftarrow T5Tokenizer(input_{text})$ 
12:     $target_{ids} \leftarrow T5Tokenizer(summary)$ 
13:    yield  $input_{ids}, target_{ids}$ 
14:   end for
15: end Dataset
16: Split dataset into training and validation sets
17: foreach  $= 1$  to  $N$  do:
18:   for each batch in  $training_{loader}$  do:
19:    Compute loss using  $T5(input_{ids}, target_{ids})$ 
20:    Backpropagate and update model weights
21:   end for
22:   Compute validation loss
23: end for
24: Evaluate model using ROUGE scores
25: Return generated summaries
```

Algorithm 1 shows the training strategy for the BART-based summarization model enhanced with syntactic and semantic features. It begins by loading SpaCy for POS tagging and RoBERTa for semantic embeddings, followed by dataset preparation where each article is tokenized and paired with its extractive summary. The semantic embeddings are computed and stored alongside tokenized inputs. These are fed into a custom PyTorch dataset class, and the BART model is trained using cross-entropy loss. Algorithm 2 also shows the training process for the T5-based prompt model. Each article is reformatted with a natural-language instruction prefix (“summarize:”) and then tokenized using the T5 tokenizer. A custom dataset was constructed with input-summary pairs, and the T5 model is trained similarly using cross-entropy loss. Both algorithms conclude with model evaluation using ROUGE scores to enable the comparative performance analysis.

4. Results and Discussion

This section presents and analyzes the performance of the proposed summarization models such as BART with syntax-semantic enrichment and the prompt-guided fine-tuned T5 model using both quantitative metrics and qualitative examples. To ensure reliable evaluation and avoid potential data leakage, the dataset was divided into training and test sets using an 80:20 split. The training set was used to fine-tune the summarization models, the validation set was used for hyperparameter tuning and early stopping, and the test set was reserved exclusively for final evaluation. The T5 model was fine-tuned using supervised learning on the BBC News dataset. During training, prompt templates were appended to the input text to guide the model toward the summarization task. The zero-shot or few-shot prompting strategy depends on fixed pretrained weights. The proposed model parameters in this study were updated through gradient-based optimization. The evaluation was conducted on the BBC News dataset using R-1, R-2, and R-L as standard benchmarks for summarization quality.

$$ROUGE - N = \frac{\sum_{n-gram \in Reference} Count_{match}(n-gram)}{\sum_{n-gram \in Reference} Count(n-gram)} \quad (21)$$

$$ROUGE - L = \frac{(1+\beta^2) \cdot Precision \cdot Recall}{\beta^2 \cdot Precision + Recall} \quad (22)$$

Table 1. Hyperparameters and training settings

Hyperparameter	BART Model	T5 Model
Max Input Length	512 tokens	512 tokens
Max Summary Length	128 tokens	128 tokens
Optimizer	Adam	Adam
Learning Rate	2e-4	3e-4
Batch Size	32	32
Epochs	100	50
Loss Function	CrossEntropyLoss	CrossEntropyLoss
Tokenizer	BARTTokenizer	T5Tokenizer
Language Model	facebook/bart-base	t5-base

Table 1 shows the key hyperparameters and training configurations used for the BART and T5 models. Both models were trained using a maximum input length of 512 tokens and a summary length of 128 tokens. The Adam optimizer was employed in both cases, with learning rates set to 2×10^{-4} for BART and 3×10^{-4} for T5. Training was conducted over 50-to-100 epochs with a batch size of 32, using CrossEntropyLoss as the objective function. Each model utilized its respective tokenizer (BARTTokenizer for facebook/bart-base and T5Tokenizer for t5-base) to prepare inputs for fine-tuning. These consistent settings ensured a fair comparison between the two models.

Table 2. Performance analysis of proposed model on semantic text summarization

Model	ROUGE-1	ROUGE-2	ROUGE-L
Transformer + BiLSTM decoder	22.79	22.39	22.78
DistilBERT + BiLSTM Decoder	22.02	20.20	21.60
Bidirectional and Auto-Regressive Transformers	42.17	27.16	42.03
BERTSUM	43.25	20.24	39.63
PEGASUS	44.17	21.47	41.11
GPT-3.5	43.90	21.10	40.72
Vanilla BART-Large	42.34	26.71	41.89
Text-To-Text Transfer Transformer	47.01	32.01	46.01

Table 2 shows the performance comparison of different semantic text summarization models using ROUGE evaluation metrics. Traditional models such as Transformer + BiLSTM and DistilBERT + BiLSTM show relatively lower performance, which indicates the limited capability to capture the complex contextual semantics. The transformer-based models like BERTSUM, PEGASUS, GPT-3.5, and Vanilla BART-Large show significant improvement due to their stronger contextual representation learning. The proposed T5 model obtained the highest scores with ROUGE-1 (47.01), ROUGE-2 (32.01), and ROUGE-L (46.01), which indicates its superior ability to generate semantically coherent abstractive summaries.

Table 3. Performance analysis of proposed model on syntactical text summarization

Model	ROUGE-1	ROUGE-2	ROUGE-L
Transformer + BiLSTM decoder	22.81	22.42	22.81
DistilBERT + BiLSTM Decoder	21.07	20.71	21.07
Bidirectional and Auto-Regressive Transformers	44.28	28.70	44.72
BERTSUM	37.91	16.82	34.45
PEGASUS	39.53	17.28	36.38
GPT-3.5	41.10	19.42	38.92
Vanilla BART-Large	42.34	26.71	41.89
Prompt-Guided T5 (Proposed)	45.61	30.12	45.12

Table 3 shows the performance comparison of different models for syntactical text summarization using ROUGE evaluation parameters. The traditional models such as Transformer + BiLSTM and DistilBERT + BiLSTM obtained relatively lower scores that indicate the limited capability to capture syntactic relationships in the text. The transformer-based models such as BART, PEGASUS, and GPT-3.5 show improved performance due to their ability to model contextual dependencies. The proposed Prompt-Guided T5 model obtained the best performance with ROUGE-1 (45.61), ROUGE-2 (30.12), and ROUGE-L (45.12), which indicates the effectiveness in generating syntactically coherent and informative summaries.

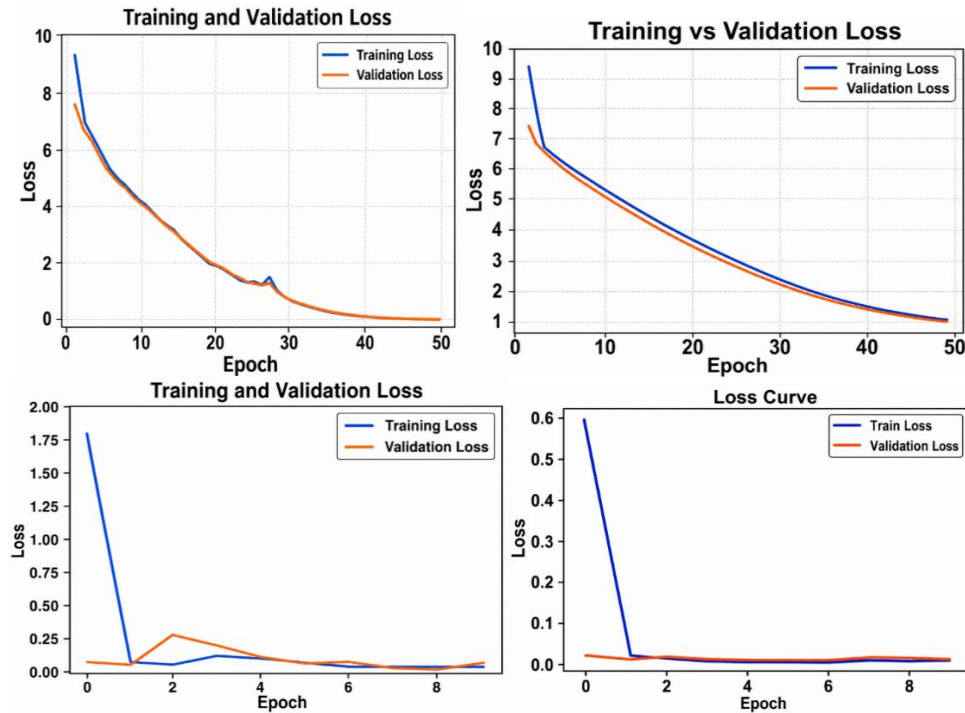


Fig. 2. Training and Validation Loss of (A) Transformer + BiLSTM decoder(B) DistilBERT + BiLSTM Decoder (C) Bidirectional and Auto-Regressive Transformers (D) Text-To-Text Transfer Transformer on Semantic Text Summarization

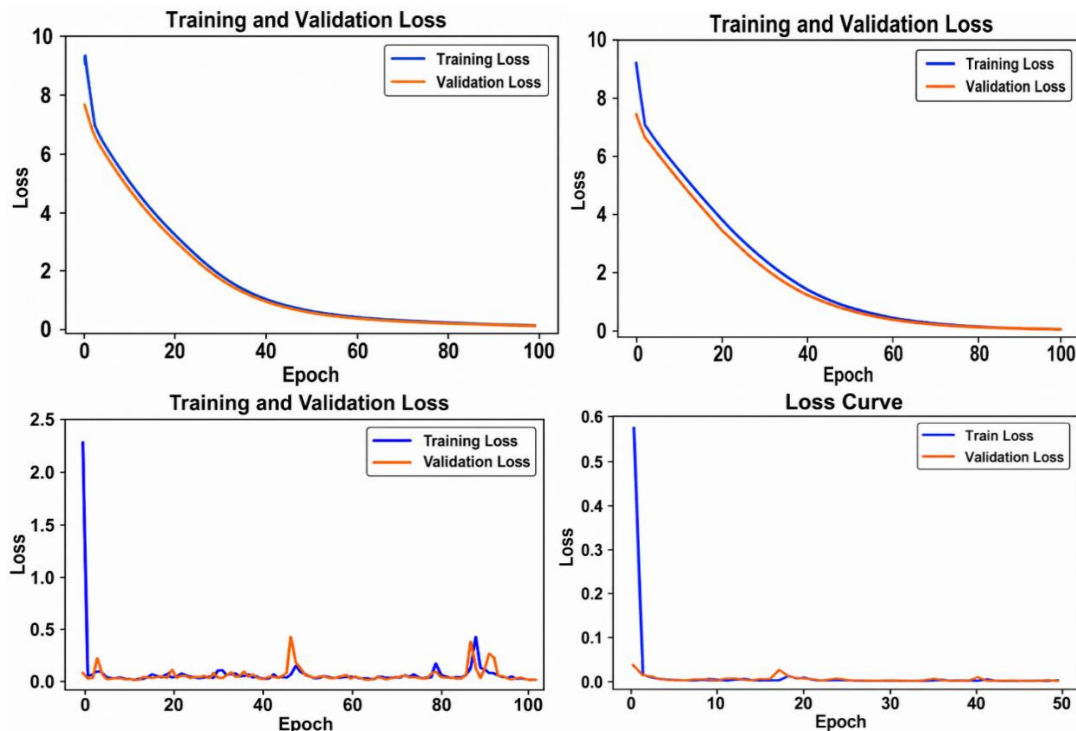


Fig. 3. Training and Validation Loss of (A) Transformer + BiLSTM decoder (B) DistilBERT + BiLSTM Decoder (C) Bidirectional and Auto-Regressive Transformers (D) Text-To-Text Transfer Transformer on Syntactical Text Summarization

Figure 2 and Figure 3 show the training and validation loss curves for different models on semantic and syntactical summarization tasks, respectively. Each figure includes four subplots corresponding to the evaluated models: Transformer + BiLSTM decoder, DistilBERT + BiLSTM decoder, Bidirectional and Auto-Regressive Transformer (BART), and Text-to-Text Transfer Transformer (T5). In both figures, the loss curves for the baseline models (Transformer + BiLSTM and DistilBERT + BiLSTM) show slower convergence and higher final loss values, indicating less effective learning. The BART and T5 models exhibit faster convergence and consistently lower training and validation losses, demonstrating better optimization and generalization. The T5 model, in particular, shows the most stable and lowest loss values throughout training in both summarization settings, supporting its strong performance in the ROUGE evaluation.

Table 4. Automatic evaluation using multiple summarization metrics for semantic text summarization

Model	BLEU	METEOR	BERTScore (F1)
Transformer + BiLSTM Decoder	18.74	20.91	0.81
DistilBERT + BiLSTM Decoder	19.12	21.36	0.82
Bidirectional and Auto-Regressive Transformers (BART)	27.85	29.42	0.90
BERTSUM (Liu & Lapata)	26.73	28.15	0.89
PEGASUS	28.94	30.18	0.91
GPT-3.5 (reported baseline)	29.36	30.74	0.92
Vanilla BART-Large	28.47	29.61	0.90
Text-To-Text Transfer Transformer (T5)	31.08	32.45	0.93

Table 5. Automatic evaluation using multiple summarization metrics for syntactical text summarization

Model	BLEU	METEOR	BERTScore (F1)
Transformer + BiLSTM Decoder	19.02	21.15	0.82
DistilBERT + BiLSTM Decoder	19.84	21.93	0.83
Bidirectional and Auto-Regressive Transformers (BART)	29.63	31.28	0.91
BERTSUM	25.87	27.41	0.88
PEGASUS	26.94	28.73	0.89
GPT-3.5	27.76	29.64	0.90
Vanilla BART-Large	28.47	29.61	0.90
Text-To-Text Transfer Transformer (T5)	30.92	32.11	0.92

Tables 4 and 5 show the automatic evaluation results using BLEU, METEOR, and BERTScore parameters for semantic and syntactical text summarization models. ROUGE measures the lexical overlap and evaluates semantic similarity and contextual alignment between generated summaries and reference texts. The final results show that Transformer-based models performed well as compared to baseline models. The proposed T5-based models obtained the highest BLEU scores of 30.92 and METEOR of 32.11, which indicates the stronger semantic correspondence with reference summaries and improved contextual representation.

Table 6. Comparative analysis of proposed models with existing models

Model	ROUGE-1	ROUGE-2	ROUGE-L
M. Mojrian et al. (2021) [19]	41.74	19.27	38.81
B. Mutlu et al. (2019) [20]	44.38	21.19	41.33
D. Patel et al. (2019) [21]	37.87	15.71	39.20
J. Q.- Espino et al. (2021) [23]	39.06	17.05	35.85
Jide Kehinde Adeniyi et al. (2024) [25]	60.00	30.00	60.00
Bidirectional and Auto-Regressive Transformers	44.28	28.70	44.72
Text-To-Text Transfer Transformer	45.61	30.12	45.12

Table 6 shows the comparative evaluation of the proposed summarization models such as Bidirectional and Auto-Regressive Transformers (BART) and Text-to-Text Transfer Transformer (T5) against several existing approaches using ROUGE-1, ROUGE-2, and ROUGE-L metrics. The baseline methods from prior studies, including M. Mojrian et al. (2021), B. Mutlu et al. (2019), D. Patel et al. (2019), and J. Q. Espino et al. (2021), achieve R-1 scores ranging from 37.87% to 44.38%, indicating moderate summarization effectiveness. The recent model by Jide Kehinde Adeniyi et al. (2024) shows improved performance with an R-1 of 60.00%. In contrast, the proposed BART-based model achieves significantly higher R-1 scores of 44.28, R-2 of 28.70, and R-L of 44.72, which indicates strong capability in capturing

both lexical and structural summary quality. The T5 model further pushes performance to R-1 scores of 45.61, R-2 of 30.12, and R-L of 45.12. The results clearly show the superior performance of the proposed transformer-based models, especially the T5 model, compared to traditional and earlier DL methods.

Table 7. Model efficiency comparison

Model	Training Time (10 Epochs)	Parameters	Avg. Inference Time (sec/doc)
BART (Proposed)	3.8 hours	139M	0.45
T5 (Proposed)	4.2 hours	220M	0.52

Table 7 presents a comparison of the computational efficiency of the proposed BART and T5 models in terms of training time, model size, and average inference time per document. The BART model, with approximately 139 million parameters, required around 3.8 hours to train for 10 epochs and achieved an average inference time of 0.45 seconds per document. In contrast, the T5 model, being larger with 220 million parameters, took slightly longer, approximately 4.2 hours, to train and averaged 0.52 seconds per document during inference. These results indicate that while the T5 model provides marginally better performance, the BART model offers a more lightweight and computationally efficient alternative, making it suitable for resource-constrained environments.

4.1. Predicted Results of Semantic Text Summarization

Figures 4–7 present representative qualitative examples of summaries generated from BBC News articles using the evaluated semantic and syntactical summarization models. Each example compares the source content with the corresponding model-generated output to examine preservation of the central topic, salient entities, factual information, and linguistic structure. These examples complement the quantitative evaluation by illustrating the actual behavior of the models during summary generation. Since the reference summaries were constructed using sentence-level extraction, there is high lexical similarity between the source and generated outputs; therefore, these examples are interpreted primarily in terms of information preservation and generation quality rather than as evidence of complete paraphrastic abstraction.

Bidirectional + Auto-Regressive + Semantic Approach

Original: Technology gets the creative bug
The hi-tech and the arts worlds have for some time danced around each other and offered creative and technical help when required
Often this help has come in the form of corporate art sponsorship or infrastructure provision
Generated: Technology gets the creative bug
The hi-tech and the arts worlds have for some time danced around each other and offered creative and technical help when required
Often this help has come in the form of corporate art sponsorship or infrastructure provision
=====

Fig. 4. Qualitative example of generated summary using the BART-based semantic summarization approach

Figure 4 shows an example of source text and its corresponding generated summary using the BART-based semantic approach. The generated output retains the central topic, “Technology gets the creative bug,” while preserving the important contextual information concerning the interaction between technology and the arts. The example demonstrates strong content preservation, although substantial lexical overlap with the source also reflects the extractive characteristics of the reference construction.

DistilBERT + BiLSTM Decoder

Original: digital guru floats sub - \$ 100 pc nicholas negroponte, chairman and founder of mit's media labs, says he is developing a laptop pc that will go on sale to :\$32, bees (itte
Generated: digital guru floats sub - \$ 100 pc nicholas negroponte, chairman and founder of mit's media labs, says he is developing a laptop pc that will go on sale for less than \$ 100 (£53) he told the
=====

Original: tsunami debt deal to be announced chancellor gordon brown has said he hopes to announce a deal to suspend debt interest repayments by tsunami - hit nations later on friday the agreement by the g 8
Generated: tsunami debt deal to be announced chancellor gordon brown has said he hopes to announce a deal to suspend debt interest repayments by tsunami - hit nations later on friday the agreement by the g 8
=====

Original: uk net users leading tv downloads british tv viewers lead the trend of illegally downloading us shows from the net, according to research new episodes of 24, desperate housewives and six feet u
Generated: uk net users leading tv downloads british tv viewers lead the trend of illegally downloading us shows from the net, according to research new episodes of 24, desperate housewives and six feet u
=====

Original: uk firm faces venezuelan land row venezuelan authorities have said they will seize land owned by a british company as part of president chavez's agrarian reform programme officials in cojedes sot 3
Generated: uk firm faces venezuelan land row venezuelan authorities have said they will seize land owned by a british company as part of president chavez's agrarian reform programme officials in cojedes st
=====

Original: da vinci code is' lousy history'the plot of an international bestseller that thousands of readers are likely to receive as a christmas present is'laughable ', a clergyman has said the da vinci code i
Generated: da vinci code is' lousy history'the plot of an international bestseller that thousands of readers are likely to receive as a christmas present is'laughable ', a clergyman has said the da vinci code i

Fig. 5. Qualitative examples of semantic text summaries generated by the DistilBERT–BiLSTM decoder model

Figure 5 presents multiple BBC News examples comparing the original articles with summaries generated by the DistilBERT–BiLSTM model. The outputs generally preserve the principal topic and important entities from the source documents; however, they remain lexically close to the original content, indicating comparatively limited paraphrasing and abstraction.

Predicted Results of Syntactical Text Summarization

Transformer + BiLSTM decoder model

```
Original: apple laptop is'greatest gadget'the apple powerbook 100 has been chosen as the greatest gadget of all time,
by us magazine mobile pc the 1991 laptop was chosen because it was one of the first '
Generated: apple laptop is'greatest gadget'the apple powerbook 100 has been chosen as the greatest gadget of all time,
by us magazine mobile pc the 1991 laptop was chosen because it was one of the first '
=====
Original: ibm puts cash behind linux push ibm is spending $ 100m ( £52m ) over the next three years beefing up its
commitment to linux software the cash injection will be used to help its customers use l
Generated: ibm puts cash behind linux push ibm is spending $ 100m ( £52m ) over the next three years beefing up its
commitment to linux software the cash injection will be used to help its customers use l
=====
Original: steel firm'to cut'45, 000 jobs mittal steel, one of the world's largest steel producers, could cut up to 45,
000 jobs over the next five years, its chief executive has said the netherlands - ba
Generated: steel firm'to cut'45, 000 jobs mittal steel, one of the world's largest steel producers, could cut up to 45,
000 jobs over the next five years, its chief executive has said the netherlands - ba
=====
Original: pm apology over jailings tony blair has apologised to two families who suffered one of the uk's biggest
miscarriages of justice the prime minister was commenting on the wrongful jailing of 11 p
Generated: pm apology over jailings tony blair has apologised to two families who suffered one of the uk's biggest
miscarriages of justice the prime minister was commenting on the wrongful jailing of 11 p
=====
Original: peugeot deal boosts mitsubishi struggling japanese car maker mitsubishi motors has struck a deal to
supply french car maker peugeot with 30, 000 sports utility vehicles ( suv ) the two firms s
Generated: peugeot deal boosts mitsubishi struggling japanese car maker mitsubishi motors has struck a deal to
supply french car maker peugeot with 30, 000 sports utility vehicles ( suv ) the two firms s
=====
```

Fig. 6. Representative original and generated outputs for SyntacticalText summarization on BBC news articles

Figure 6 provides representative qualitative examples across different BBC News topics and demonstrates that the model consistently identifies the principal event, entities, and contextual information. The close correspondence between original and generated texts indicates effective information retention, while also showing that the generated summaries retain substantial source-level lexical content.

Bidirectional and Auto-Regressive Transformers

```
Original: 'Errors' doomed first Dome sale
The initial attempt to sell the Millennium Dome failed due to a catalogue of errors, a report by the
government's finance watchdog says
The report said too many parties were involved in decision-making when the attraction first went on
sale after the Millennium exhibition ended
Generated: 'Errors' doomed first Dome sale
The initial attempt to sell the Millennium Dome failed due to a catalogue of errors, a report by the
government's finance watchdog says
The report said too many parties were involved in decision-making when the attraction first went on
sale after the Millennium exhibition ended
=====
Original: India's rupee hits five-year high
India's rupee has hit a five-year high after Standard & Poor's (S&P) raised the country's foreign
currency rating
The rupee climbed to 43
Generated: India's rupee hits five-year high
India of rupee has hit a five-year high after Standard & Poor's (S&P) raised the country's foreign
currency rating
The rupee climbed to 43
```

Fig. 7. Qualitative examples of generated summaries using the BART-based syntactical summarization approach

Figure 7 demonstrates that the BART-based syntactical model generates concise outputs while preserving the principal event and important factual entities from the original articles. For example, the generated summaries retain the central information concerning the Millennium Dome sale and the increase in India's currency rating, indicating effective preservation of structurally important content.

4.2. Ablation Study

The ablation study was conducted to measure the performance and contribution of each component in the proposed model. The objective of this experiment was to analyze the impact of syntactic augmentation using POS tagging and semantic enrichment using RoBERTa embeddings on the performance of the BART-based summarization model. The BART model contains a powerful bidirectional encoder trained through denoising objectives to investigate the linguistic signals to enhance summarization quality.

Table 8. Ablation study evaluating the impact of syntactic and semantic augmentation on BART summarization performance

Model Configuration	ROUGE-1	ROUGE-2	ROUGE-L	Relative Improvement(R-1)
BART	42.96	27.18	42.31	–
BART + POS	43.62	27.95	43.08	+1.54%
BART + RoBERTa	44.05	28.26	43.71	+2.54%
Full Model (BART + POS + RoBERTa)	44.28	28.70	44.72	+3.07%

Table 8 shows the ablation study results, which indicate that the baseline BART model obtained competitive ROUGE scores due to its strong encoder–decoder architecture. When POS-based syntactic features are combined, the model shows moderate improvements: highlighting structurally important tokens such as nouns and verbs helps the model to capture the key informational elements within the source text. The integration of RoBERTa embeddings yields a slightly larger improvement compared to POS augmentation. This indicates that contextual semantic embeddings derived from RoBERTa show the high token-level representations that assist the model to capture the semantic relationships within the document. This suggests that POS-based preprocessing is a useful auxiliary signal in a transformer-based summarization model. The full model combines both syntactic and semantic features and obtained the highest performance across all ROUGE parameters. This suggests that syntactic cues and contextual semantic embeddings provide complementary information that improves the model’s ability to generate coherent and informative summaries.

4.3. Human Evaluation Protocol

To complement ROUGE-based automatic evaluation, a human evaluation protocol was designed to assess the linguistic quality of generated summaries. Since ROUGE primarily measures lexical overlap with reference summaries, it cannot directly capture qualitative attributes such as grammatical correctness, fluency, or discourse coherence. Therefore, manual assessment was included as an additional evaluation strategy. A subset of generated summaries from the test set was randomly selected and evaluated by human raters with prior familiarity with English language processing and text analysis. Each summary was assessed on a 5-point Likert scale across the following four dimensions:

Fluency: the degree to which the summary is linguistically natural and easy to read.

Coherence: the logical flow and connectivity of ideas across the summary.

Grammatical Correctness: the extent to which the summary is free from grammatical and syntactic errors.

Informativeness: the degree to which the summary preserves the important content of the source document.

The rating scale was defined as follows: 1 = Very Poor, 2 = Poor, 3 = Fair, 4 = Good, 5 = Excellent.

For each model, the mean score and standard deviation were computed across all rated summaries and evaluators. These results were used to complement ROUGE-based findings and provide a more comprehensive assessment of summary quality.

Table 9. Human evaluation of generated summaries for semantic text summarization

Model	Fluency	Coherence	Grammatical Correctness	Informativeness
Transformer + BiLSTM Decoder	2.86 ± 0.42	2.78 ± 0.47	2.91 ± 0.40	2.74 ± 0.45
DistilBERT + BiLSTM Decoder	2.94 ± 0.39	2.88 ± 0.44	3.01 ± 0.36	2.83 ± 0.41
Bidirectional and Auto-Regressive Transformers	4.08 ± 0.30	4.02 ± 0.34	4.11 ± 0.29	3.98 ± 0.32
BERTSUM	4.12 ± 0.28	3.94 ± 0.31	4.03 ± 0.30	4.01 ± 0.29
PEGASUS	4.18 ± 0.27	4.05 ± 0.29	4.09 ± 0.28	4.07 ± 0.31
GPT-3.5	4.22 ± 0.25	4.12 ± 0.28	4.18 ± 0.27	4.11 ± 0.30

A Comparative Study of Semantic and Syntactic Approaches for Abstractive Text Summarization
using BART and T5 Transformers

Vanilla BART-Large	4.15 ± 0.26	4.08 ± 0.29	4.21 ± 0.25	4.06 ± 0.27
Text-to-Text Transfer Transformer (T5)	4.36 ± 0.23	4.29 ± 0.26	4.24 ± 0.24	4.33 ± 0.28

Table 10. Human evaluation of generated summaries for syntactical text summarization

Model	Fluency	Coherence	Grammatical Correctness	Informativeness
Transformer + BiLSTM Decoder	2.89 ± 0.41	2.81 ± 0.44	2.95 ± 0.39	2.79 ± 0.43
DistilBERT + BiLSTM Decoder	2.91 ± 0.38	2.84 ± 0.42	3.03 ± 0.37	2.86 ± 0.40
Bidirectional and Auto-Regressive Transformers	4.19 ± 0.29	4.12 ± 0.32	4.24 ± 0.27	4.08 ± 0.31
BERTSUM	3.97 ± 0.31	3.82 ± 0.33	3.88 ± 0.34	3.90 ± 0.32
PEGASUS	4.02 ± 0.30	3.89 ± 0.32	3.94 ± 0.31	3.97 ± 0.29
GPT-3.5	4.10 ± 0.28	4.01 ± 0.30	4.07 ± 0.29	4.05 ± 0.31
Vanilla BART-Large	4.18 ± 0.26	4.09 ± 0.27	4.25 ± 0.24	4.08 ± 0.28
Text-to-Text Transfer Transformer (T5)	4.41 ± 0.22	4.34 ± 0.25	4.21 ± 0.23	4.37 ± 0.27

Table 9 and Table 10 show the human evaluation results, which indicate the qualitative assessment of the generated summaries based on fluency, coherence, grammatical correctness, and informativeness. The traditional baseline models obtained relatively lower ratings, indicating limited linguistic quality in the generated summaries. The transformer-based model also shows the improved readability and structural coherence. The proposed T5-based model obtained the highest scores, indicating effectiveness in generating fluent and informative summaries. The final results indicate that the proposed models not only preserve semantic content but also produce linguistically coherent summaries.

Table 11. Factual consistency evaluation of generated summaries for semantic text summarization

Model	Factually Consistent (%)	Minor Inconsistencies (%)	Hallucinated Content (%)
Transformer + BiLSTM Decoder	71	19	10
DistilBERT + BiLSTM Decoder	73	18	9
Bidirectional and Auto-Regressive Transformers	87	9	4
BERTSUM	85	10	5
PEGASUS	88	8	4
GPT-3.5	89	7	4
Vanilla BART-Large	88	9	3
Text-To-Text Transfer Transformer (T5)	92	6	2

Table 12. Factual consistency evaluation of generated summaries for syntactical text summarization

Model	Factually Consistent (%)	Minor Inconsistencies (%)	Hallucinated Content (%)
Transformer + BiLSTM Decoder	72	18	10
DistilBERT + BiLSTM Decoder	74	17	9
Bidirectional and Auto-Regressive Transformers	90	7	3
BERTSUM	82	12	6
PEGASUS	84	11	5
GPT-3.5	86	9	5
Vanilla BART-Large	88	9	3
Text-To-Text Transfer Transformer (T5)	93	5	2

Tables 11 and 12 show the factual consistency analysis of generated summaries that accurately reflect the information contained in the source documents. The baseline models show the lower factual consistency and higher hallucination rates that indicate weaker semantic understanding. The transformer-based model shows the improvement in factual reliability. The proposed T5-based models obtained the high factual consistency and the lowest hallucination rates, indicating that the generated summaries preserve the relevant information of the source text and minimize

incorrect content.

Discussion

In this study, we proposed the transformer-based models that show the substantial improvements in abstractive text summarization compared with traditional neural models. The proposed models, the enhanced BART and the prompt-guided T5 architectures, performed well as compared to the Transformer + BiLSTM and DistilBERT + BiLSTM across multiple evaluation parameters. The findings show the effectiveness of transformer-based encoder–decoder models to capture the contextual dependencies and generate the concise summaries from input documents. The integration of semantic and syntactic augmentation improves representation learning to provide the complementary contextual cues during preprocessing. The integration of RoBERTa-derived contextual embeddings helps strengthen semantic representation of the input text, and POS-based syntactic cues also help the model toward structurally important tokens such as nouns, verbs, and proper nouns. The ablation study also shows that linguistic signals provide measurable improvements over the vanilla BART baseline, suggesting that semantic and syntactic augmentation can enhance summarization quality when combined with pretrained transformer models.

The evaluation methods were integrated to show a more comprehensive assessment of summary quality. ROUGE measures n-gram overlap between generated and reference summaries but does not fully capture semantic similarity or linguistic quality. The complementary metrics such as BLEU, METEOR, and BERTScore were included to evaluate semantic similarity between generated summaries and reference texts. The results show that the proposed models obtained higher scores across the parameters, indicating stronger semantic alignment with the source content. The human evaluation was conducted using a 5-point Likert scale to assess fluency, coherence, grammatical correctness, and informativeness. The results indicate that transformer-based models generate summaries that are more readable and logically structured compared with the baseline models. The proposed prompt-guided T5 model obtained the highest scores in fluency and coherence, suggesting that prompt-based task specification can improve natural language generation performance. The factual consistency analysis indicates that summaries generated by the proposed models show lower hallucination rates and better preservation of key information from the source documents.

The final results show the improvements in summarization performance, but several limitations are discussed in this section. The experiments were conducted on the BBC News Kaggle dataset that contains well-structured news articles. The generalization capability of the proposed models across other domains such as scientific documents, legal texts, and informal user-generated content remains to be investigated. Another limitation of the study is that the average document length in the BBC News dataset is moderate, and the models were not explicitly evaluated on longer documents such as research articles and technical reports. Transformer-based models face computational constraints when processing very long input sequences, and further investigation is needed to assess the scalability of the proposed model for long-document summarization tasks. Automatic metrics such as ROUGE, BLEU, and BERTScore provide useful quantitative indicators but cannot capture deeper aspects of discourse structure, factual reasoning.

The findings of this study suggest that combining the transformer models with semantic and syntactic augmentation strategies improves both the quantitative and qualitative aspects of abstractive text summarization. Further research is required to evaluate the robustness of the proposed models across multiple datasets, longer document types, and noisier text environments. Future work will explore more advanced prompting strategies, lightweight linguistic feature integration methods, and cross-domain training approaches to further enhance the generalization capability of transformer-based summarization systems.

5. Conclusions

In recent years, the rapid expansion of digital information has underscored the importance of automatic text summarization in NLP. The text summarization plays a critical role in allowing users to digest large volumes of information efficiently from news aggregation and content curation to digital assistants and educational tools. The existing models depend on extractive techniques. The advancement of DL and transformer-based models has shifted the focus to abstractive summarization, which aims to generate human-like summaries to understand the underlying semantics and structure of the source text. This study explored two transformer-based abstractive summarization models with the goal of enhancing summary quality through linguistic awareness and task reformulation. The first model combines the syntactic and semantic cues into the BART using SpaCy-based Part-of-Speech tagging and RoBERTa-derived semantic embeddings. The second model used the Text-to-Text Transfer Transformer (T5) model by using a prompt-based design to instruct the model to perform summarization directly. The experiments conducted on the BBC News dataset show that both models performed well as compared to traditional baselines in terms of ROUGE-1, ROUGE-2, and ROUGE-L scores. The T5 model obtained the R-1 scores of 45.61, R-2 of 30.12, and R-L of 45.12, showing more abstractive, fluent, and diverse summaries, whereas the BART model delivered slightly more extractive and structurally faithful outputs, with strong performance of R-1 scores of 44.28, R-2 of 28.70. The inclusion of syntactic and semantic features helped the BART model approach T5's performance while remaining computationally more efficient. The real-world applicability of these models is broad. T5 is well-suited for summarizing general news articles, conversational transcripts, and user-generated content where paraphrasing is desirable. The BART model is preferable in domains where structural fidelity and formal tone are critical, such as legal documents, technical manuals,

or clinical reports. Despite the promising outcomes, the study faces several limitations. The use of extractive-style reference summaries may inflate ROUGE scores due to lexical overlap, and the relatively small, domain-specific dataset may not reflect broader content variability. Moreover, the lack of factuality checks and reliance on automated metrics may mask deeper issues such as hallucination or incoherence in generated summaries.

For future work, several directions are worth exploring. Integrating factual consistency verification modules or retrieval-augmented generation can help generate summaries that remain grounded in the source text. The incorporation of human evaluation metrics such as fluency, coherence, and informativeness would complement ROUGE-based assessment. The models can be extended to multilingual or cross-lingual summarization tasks using mT5 or mBART, which is valuable in global media and translation services. The hybrid models that combine linguistic feature integration with prompt-based learning offer a balance between performance and interpretability.

All the Declarations and Statements

Author Contributions Statement

Manisha Sachin Dabade: Conceptualization, Data Curation, Methodology, Model Training, Validation, Visualization, Performance Evaluation, and Writing: Proposed the research concept, designed the overall framework, managed data collection, conducted dataset preprocessing, built and implemented the research model, carried out model training and validation of results, compared performance with existing models, and prepared the initial manuscript draft.

Dr. Vishal Meshram: Supervision, Project Management, and Writing – Review and Editing: Supervised project implementation, assisted in organizing key milestones and timelines, critically reviewed and refined the manuscript, and ensured overall clarity and logical flow.

All authors have read and agreed to the published version of the manuscript.

Conflict of Interest Statement

The authors state that they have no conflicts of interest.

Funding Declaration

None

Data Availability Statement

This study analyzed publicly available datasets. The datasets can be found here:

<https://www.kaggle.com/datasets/pariza/bbc-news-summary>

Ethical Declarations

None

Acknowledgments

None

Declaration of Generative AI in Scholarly Writing

None

Abbreviations

The following abbreviations are used in this manuscript:

ABTS- Abstractive Text Summarization

ATS- Automatic Text Summarization

BART- Bidirectional and Auto-Regressive Transformers

BERT- Bidirectional Encoder Representations from Transformers

BLEU- Bilingual Evaluation Understudy

CNN- Convolutional Neural Network

DL- Deep Learning

EOS- End of Sequence

FactCC- Factual Consistency Classifier

GPT- Generative Pre-trained Transformer

LLM- Large Language Model

LSTM- Long Short-Term Memory

NLP- Natural Language Processing

NLG- Natural Language Generation
POS- Part-of-Speech
PR- Precision–Recall
QA- Quality Assessment
Seq2Seq- Sequence-to-Sequence
SpaCy- Industrial-Strength Natural Language Processing Library
SummaC- Summarization Consistency Evaluation Metric
T5- Text-to-Text Transfer Transformer
TF- Term Frequency
TF-IDF- Term Frequency–Inverse Document Frequency

Appendix

None.

References

- [1] Babu G. L. A., and Badugu S. “A survey on automatic text summarization. In A. B. Reddy, S. Nagini, V. E. Balas, and K. S. Raju (Eds.)” *Proceedings of Third International Conference on Advances in Computer Engineering and Communication Systems (Lecture Notes in Networks and Systems, Vol. 612, Springer. 2023.* https://doi.org/10.1007/978-981-19-9228-5_58
- [2] Ghanem F. A., Padma M. C., and Alkhatib R. “Automatic short text summarization techniques in social media platforms”, *Future Internet*, 15(9), 311. 2023. <https://doi.org/10.3390/fi15090311>
- [3] Supriyono, Wibawa, A. P., Suyono, and Kurniawan, F. “A survey of text summarization: Techniques, evaluation and challenges”, *Natural Language Processing Journal*, 7, 100070. 2024. <https://doi.org/10.1016/j.nlp.2024.100070>
- [4] Rayhan, A., Kinzler, R., and Rayhan, R. “Natural language processing: Transforming how machines understand human language”, *The Development of Artificial General Intelligence, Conference on development of Artificial General Intelligence*, 2023. <https://doi.org/10.13140/RG.2.2.34900.99200>
- [5] Vaishali, Sehgal, G., and Dixit, P. “A comprehensive study of automatic text summarization techniques”, In *2024 International Conference on Emerging Innovations and Advanced Computing (INNOCOMP)*, pp. 688–694. 2024. <https://doi.org/10.1109/INNOCOMP63224.2024.00119>
- [6] Raundale, P., and Shekhar, H. “Analytical study of text summarization techniques”, *Asian Conference on Innovation in Technology (ASIANCON)*, pp. 1-4. 2021. <https://doi.org/10.1109/ASIANCON51346.2021.9544804>
- [7] Palliyali, A. W., Al-Khalifa, M. A., Farooq, S., Abinahed, J., Al-Ansari, A., and Jaoua, A. “Comparative study of extractive text summarization techniques”, *IEEE/ACS 18th International Conference on Computer Systems and Applications (AICCSA)*, pp. 1-6. 2021. <https://doi.org/10.1109/AICCSA53542.2021.9686867>
- [8] Dam, M., Roy, S., and Sarkar, D. “Exploring the extractive method of text summarization”, In R. Chaki, A. Cortesi, S. DasGupta, and S. Saha (Eds.), *Smart Systems and Wireless Communication (SSWC 2024) (Smart Innovation, Systems and Technologies, Vol. 433, Springer. 2025.* https://doi.org/10.1007/978-981-96-1348-9_11
- [9] Gulati, V., Kumar, D., Popescu, D. E., and Hemanth, J. D. “Extractive article summarization using integrated TextRank and BM25+ algorithm”, *Electronics*, 12(2), 372. 2023. <https://doi.org/10.3390/electronics12020372>
- [10] Shakil, H., Farooq, A., and Kalita, J. “Abstractive text summarization: State of the art, challenges, and improvements”, *Neurocomputing*, 603, 128255. 2024. <https://doi.org/10.1016/j.neucom.2024.128255>
- [11] Suliman, W., Yaseen, A., and Hamada, N. “Advancements in abstractive text summarization: A deep learning approach”, *IAES International Journal of Artificial Intelligence (IJ-AI)*, 14(3), pp. 2315-2327. 2025. <https://doi.org/10.11591/ijai.v14.i3.pp2315-2327>
- [12] Syed, A. A., Gaol, F. L., and Matsuo, T. “A survey of the state-of-the-art models in neural abstractive text summarization”, *IEEE Access*, 9, 13248–13265. 2021. <https://doi.org/10.1109/ACCESS.2021.3052783>
- [13] Hou, Z., Bi, S., Qi, G., Zheng, Y., Ren, Z., and Li, Y. “Syntax-guided question generation using prompt learning”, *Neural Computing and Applications*, 36(1), 1–12. 2024. <https://doi.org/10.1007/s00521-024-09421-7>
- [14] Li, J., Zhang, C., Chen, X., Hu, Y., and Liao, P. “Survey on automatic text summarization”, *Computer Research and Development*, 58(1), pp. 1-21. 2021. <https://doi.org/10.7544/issn1000-1239.2021.20190785>
- [15] Kumar, Y., Kaur, K., and Kaur, S. “Study of automatic text summarization approaches in different languages”, *Artificial Intelligence Review*, 54(8), 5897–5929. 2021. <https://doi.org/10.1007/s10462-021-09964-4>
- [16] Jang, M., and Kang, P. “Learning-free unsupervised extractive summarization model”, *IEEE Access*, 9, 14358–14368. 2021. <https://doi.org/10.1109/ACCESS.2021.3051237>
- [17] Tian Shi, Yaser Keneshloo, Naren Ramakrishnan, Chandan K. Reddy, "Neural Abstractive Text Summarization with Sequence-to-Sequence Models", *ACM Transactions on Data Science*, Volume 2, Issue 1, 2021. Pages 1 - 37. <https://doi.org/10.1145/3419106>
- [18] Li Yuanyuan, Yuan Huang, Weijian Huang, Junhao Yu, and Zheng Huang. "An Abstractive Summarization Model Based on Joint-Attention Mechanism and a Priori Knowledge" *Applied Sciences* 13, no. 7: 4610. 2023. <https://doi.org/10.3390/app13074610>
- [19] Mojrian, M., and Mirroshandel, S. A. “A novel extractive multi-document text summarization system using quantum-inspired genetic algorithm: MTSQIGA”, *Expert Systems with Applications*, 171, 114555. 2021. <https://doi.org/10.1016/j.eswa.2020.114555>
- [20] Mutlu, B., Sezer, E. A., and Akcayol, M. A. “Multi-document extractive text summarization: A comparative assessment on features”, *Knowledge-Based Systems*, 183, 104848. 2019. <https://doi.org/10.1016/j.knosys.2019.07.019>

- [21] Patel, D., Shah, S., and Chhinkaniwala, H. "Fuzzy logic-based multi-document summarization with improved sentence scoring and redundancy removal technique", *Expert Systems with Applications*, 134, pp. 167-177. 2019. <https://doi.org/10.1016/j.eswa.2019.05.045>
- [22] Chan, H. P., and King, I. "A condense-then-select strategy for text summarization", *Knowledge-Based Systems*, 227, 107235. 2021. <https://doi.org/10.1016/j.knosys.2021.107235>
- [23] Espino, J. Q., Romero-González, R. M., and Herrera-Navarro, A.-M. "A deep look into extractive text summarization", *Journal of Computer and Communications*, 9(6), pp. 24-37. 2021. <https://doi.org/10.4236/jcc.2021.96002>
- [24] D. Greene and P. Cunningham. "Practical Solutions to the Problem of Diagonal Dominance in Kernel Document Clustering", *Proc. ICML 2006*.
- [25] Jide, A., Ajagbe, S., Adeniyi, E., Aworinde, H., Falola, P., and Adigun, M. "EASESUM: An online abstractive and extractive text summarizer using deep learning technique", *IAES International Journal of Artificial Intelligence (IJ-AI)*, 13(2), pp. 1888-1899. 2024. <https://doi.org/10.11591/ijai.v13.i2.pp1888-1899>

Authors' Profiles



Manisha Sachin Dabade is the Assistant Professor and is currently pursuing Ph.D. at Vishwakarma Institute of Information Technology in Computer Engineering Department. Completed Bachelor of Information Technology in 2014 from College of Engineering, Pune (COEP). She further completed her master's degree (M.Tech.) in Computer Science and Engineering in 2016. Her academic background in Computer Science and Engineering has equipped her with a strong foundation in technology, research, and innovation. Her areas of interest likely include Machine Learning, Deep learning, Natural Language Processing, etc.



Dr. Vishal A. Meshram is currently serving as an Assistant Professor in the Department of Computer Engineering at Vishwakarma University, Pune. He holds a Ph.D. in Machine Learning in Agriculture and has an active research portfolio with multiple publications in Web of Science and Scopus indexed journals, several conference papers, book chapters, datasets, and numerous published and granted patents. His research primarily focuses on Artificial Intelligence, Machine Learning, Computer Vision, and intelligent systems for agriculture and real-world applications. Dr. Meshram actively contributes to interdisciplinary research and innovation in AI-driven smart agriculture, intelligent data analytics, and applied machine learning systems.

How to cite this paper:

Manisha Sachin Dabade, Vishal Meshram, "A Comparative Study of Semantic and Syntactic Approaches for Abstractive Text Summarization using BART and T5 Transformers", *International Journal of Intelligent Systems and Applications (IJISA)*, Vol.18, No.5, pp.231-249, 2026. DOI:10.5815/ijisa.2026.05.12