World Journal of Engineering Research and Technology

www.wjert.org

Impact Factor: 7.029 Coden USA: WJERA4

PERFORMANCE ANALYSIS OF NEURAL LANGUAGE MODELS FOR AUTOMATED TEXT SUMMARIZATION

***[1]G. Krishnaveni, [2]B. Sarushma, [3]L. Samuel Raju, [4]A. Rakesh, [5]B. Sai Pavan**

*[1]Assistant Professor Department of Information Technology, Sir C R Reddy College of Engineering, Vatluru, Andhra Pradesh, India.

[2,3,4,5]Department of Information Technology, Sir C R Reddy College of Engineering, Vatluru, Andhra Pradesh, India.

Article Received on 05/04/2025 **Article Revised on 25/04/2026** **Article Published on 04/05/2026**

***Corresponding Author**

G. Krishnaveni

Assistant Professor Department of Information Technology, Sir C R Reddy College of Engineering, Vatluru, Andhra Pradesh, India.

https://doi.org/10.5281/zenodo.20021540

How to cite this Article: *[1]G. Krishnaveni, [2]B. Sarushma, [3]L. Samuel Raju, [4]A. Rakesh, [5]B. Sai Pavan. (2026). Performance Analysis Of Neural Language Models For Automated Text Summarization. World Journal of Engineering Research and Technology, 12(5), 270–281.

ABSTRACT

The massive increase in digital text has created a huge need for effective, automated summarization systems. Manually summarizing large documents takes significant time and is often not practical. Recent improvements in transformer neural language models have helped with abstractive summarization significantly. In this research, we assess the relative performance of three neural language models for automatic text summarization, specifically BART, T5, and GPT, using the same conditions for each model (compression ratio for conciseness and ROUGE metrics for summary quality). We propose a combined performance metric to compare this trade-off of length versus informativeness. Our experimental results demonstrate encoder-decoder architectures outperform autoregressive architectures in automatic summarization tasks. BART produced the highest quality summaries among all three models evaluated. Our study proposes an evaluation framework and methods for model selection to assist in the efficient development of practical summarization systems.

KEYWORDS: Text Summarization, Neural Language Models, Metrics, Performance Analysis.

I. INTRODUCTION

The proliferation of digital information due to the internet, increased access to research publications and news outlets has left individuals unable to process an overwhelming number of words in a short period of time. Automated text summarization helps alleviate this burden by creating shorter forms of writing that still retain the most crucial aspects of the content produced in the source documents. Early automated text summarization methods were primarily based on extractive summarization, selecting sentences based upon their statistical properties without regard for the meaning of the sentences. This evolved with the advent of deep learning and the ability of transformer-based neural language models (e.g. BART, T5, and GPT) to capture contextual relationships between words in text. While these models range dramatically in performance and architecture, the objective of this research is to systematically evaluate and compare various neural language models to identify the best method for automated text summarization.

A. Problem Identification

With the enormous amount of textual content generated on a daily basis in areas such as news, education and business, users are inundated with information. Reading and summarizing lengthy documents by hand in real world scenarios can be incredibly time consuming and simply not practical. The performance of the numerous neural language models extensively used for automatic text summarization also varies widely when applied in practice. In addition, many summarization systems do not provide a balanced evaluation that allows organizations to consider summary quality as well as how brief or concise the generated summary is when selecting the best model for use on actual summarization tasks.

B. OBJECTIVE AND MOTIVATION

Our objectives are twofold:

1. To develop a neural language model-based text summarization system through automated processes such as BART, T5, and GPT.
2. To evaluate the performance of the models against each other based on their compression ratios and through the ROUGE metric.
3. To determine which of the developed models will produce accurate and concise summaries that can be used in practical applications.

This paper is driven by the fact that automatic text summarization can provide users with shorter, more meaningful summaries. Although there are a number of neural models to perform automatic text summarization (ATS), it is uncertain which neural model will work best in a real-world context to provide concise and accurate summaries, thus aiding the user to access and comprehend the information more quickly and efficiently.

II. LITERATURE REVIEW

This section presents important research on neural language models that focus on automated text summarization. We have collated the most meaningful approaches, findings, and limitations related to the use of neural language models in an effort to show how our work adds value and fills in the gaps in the literature.

[1] The evaluation of large language models (LLMs) for summarization

Nedashkovskaya and Yeremichuk (2025) conducted a study using two popular summarization datasets, XSum and CNN/DailyMail, to evaluate the efficacy of quantized versus non-quantized large language models for summarization. Their research was limited in terms of both size (i.e., they used a limited number of datasets) and scope (i.e., they had to balance accuracy and efficiency). In addition, they performed a scalability analysis on the two large language models, but the scope of their study was further limited due to the fact that both datasets do not allow for multiple datasets to be assessed simultaneously.

[2] The use of LLMs for domain-specific summarization Houamengni and Gedikli (2025) used LLMs to automate news summarization related to risks in the supply chain. Although their study was limited in that it only included a single domain and a small number of models, they evaluated the readability of their summarized content as well as how well the automatic summarization identified risk.

[3] Tools for extractive and abstractive summarization Sorokina (2024) analysed a number of AI tools that can be used for summarization, including QuillBot, WordTune, Scholarcy, and ChatGPT, and their respective capabilities to perform extractive or abstractive summarization of text. Due to the survey nature of the study, little quantitative analysis was performed on the datasets included in the survey.

[4] Conducting a Comparison of Different Models Built Using the Transformer Architecture.

In the study conducted by Chen, (2023) ROUGE and BERTScore were used as evaluation metrics on the CNN/DailyMail dataset to evaluate the performance of three models based on the Transformer architecture: GPT-3.5, Pegasus-large, and Flan-T5. While this study did not focus on either efficiency or compression, it does demonstrate that encoder-decoder Transformer architectures can achieve excellent performance when assessing performance based upon either ROUGE or BERTScore.

[5] Research on Abstractive Summarization in Specific Languages

Baykara & Güngör (2022) completed their investigation of the efficacy of pre-trained Transformer models (specifically BART and T5) for producing abstractive summaries in Turkish. Although they achieved positive outcomes through evaluation, their study is limited to the confines of one language and did not provide a cross-lingual evaluation.

[6] Synthesis of Research & Identification of Research Holes

Overall, research has produced evidence that supports the use of transformer-based architectures for text summarization. However, the research that has been conducted often focuses on a specific dataset, domain, or metric for assessment.

The current study's performance-based analysis is motivated by the absence of research that strikes a fair balance between quality of summary and conciseness when assessing performance with respect to ROUGE metrics; specifically, by using a standard comparison of multiple transformer architectures summarized into one summary with respect to different datasets of varying compression ratios.

Table 1: Summarized Review of Literature.

Year	Author(s)	Study Focus	Limitations
2025	P. Sai Viswanath & N. Manohar	Transformer-based summarization models were evaluated using BLEU, ROUGE, and BERTScore on benchmark datasets.	Performance depends on input quality, struggles withcomplex conversati - onal nuances, and requires high computational resources.
2025	Nedashkovskaya, N.I.,& Yeremichuk, R. I.	LLMs, both quantized and non-quantized, were evaluated for text summarization using BLEU, ROUGE, and BERTScore on benchmark datasets.	Limited datasets were available and there were no real-time evaluations made and there are few evaluation criteria
2025	Houamengni, L. R. P., & Gedikli, F.	The purpose of using LLMs is to automate news summarization	Limited and domain-specific evaluations of the models

		for supply chain risk analysis.	used.
2024	Sorokina, S.	Overview of AI-based extractive and abstractive text summarization methods, including commonly used techniques in both approaches.	Review process with no detailed quantitative analysis conducted..
2023	Chen, S.	Comparison of Transformer based summarization models using both ROUGE and BERTScore as there are no common evaluation criteria.	One dataset was evaluated and efficiency analyses of the model have been limited.
2022	Baykara, B., & Güngör, T	Turkish text summarization using transformer models pre-trained on abstractive summarization tasks.	Limited cross-linguistic evaluation with no generalizable conclusions.

III. DATASET AND PREPROCESSING

A. Dataset Construction

This research uses input plain text documents that have been sent to the Automatic Text Summary (ATS) system for use in creating summaries of those documents, using the BART, T5, or GPT models. Input plain text documents could be derived from a variety of sources, such as news, academic papers, and standard plain text documents.

There is currently no dataset created for any of these summarization systems (model evaluations) with input text documents that could be used to directly compare BART to TAW, or to TAW to GPT. Therefore, the research uses the same data for each of the models (existing pre-trained models); therefore, the results of the research can be compared directly.

The fact that the same input text documents are provided to all three pre-trained models means that researchers can compare the models based on their ability to create summaries of input documents. Thus, this research will provide consistent performance summaries across all three pre-trained models and thus will provide consistent performance each time input documents are processed and summarized.

B. Pre-processing Framework

- Collecting raw text inputs (provided by users) in their original unformatted Plain Text Format
- Normalising text (by removing extraneous white-space and / or other inconsistencies in formatting)

- Segmenting sentences for estimating length; structuring text for natural language processing using the Natural Language Toolkit (`NLTK); and
- Checking that length of inputs complies with token limit(s) associated with particular Language Model(s)
- Truncating input where required so that it meets the requirements of transformer-based model(s)
- Sending pre-processed text consistently as input(s) to all neural-language model(s) for summary generation

IV. PROPOSED METHODOLOGY

A. System Overview

BART, T5, and GPT will be used to summarize an original document (D) into a machine-generated summary by each model. Each model will have a summary generated independently of the others. Each model will process the same input document so that it is fair. The output summaries generated by each model will then be measured against ROUGE metrics as well as compression rate; a combined score of each model's output/summary will be produced to determine which is the highest performing model.

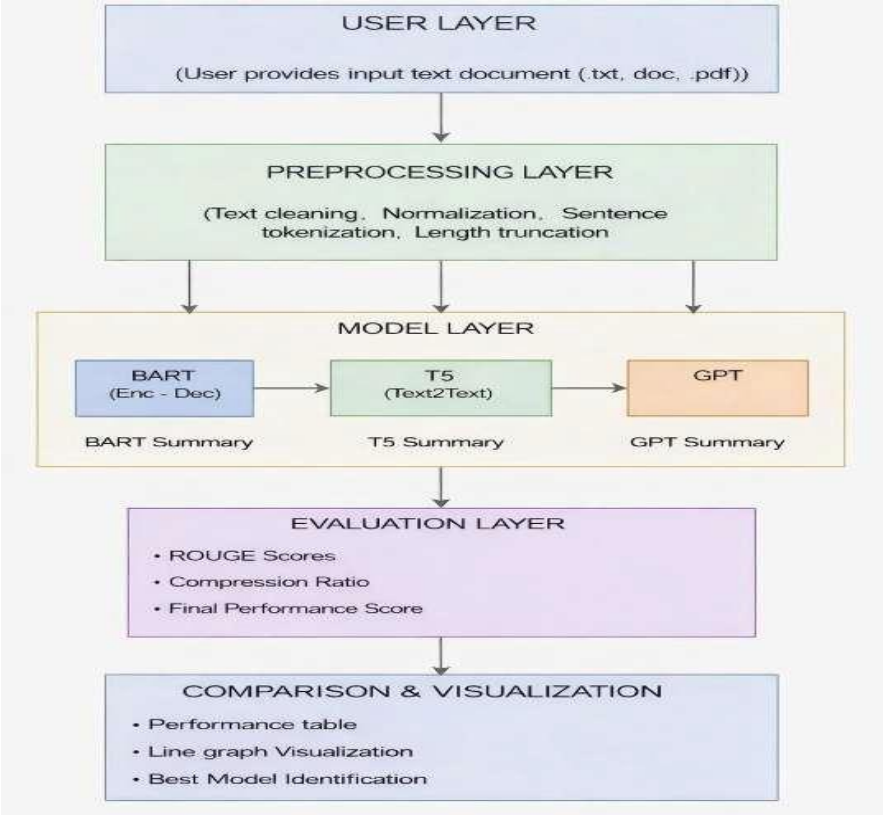

Figure I – Workflow Architecture.

A. Feature Definition

Both input and evaluation levels are used to define features in this study:

- Features of the input:
  - User-provided raw textual content.
  - Sentence structure tokenization.
  - Word count and document length.

- Features of the evaluation:
  - ROUGE-1 (unigram overlap).
  - ROUGE-2 (overlapping bigram).
  - The longest common subsequence is ROUGE-L.
  - Compression ratio: the length of the summary divided by the length of the original text.
  - Total performance score (compression ratio less average ROUGE F1).

Semantic accuracy and summary conciseness are captured by these characteristics taken together.

B. Modeling Choices

Three neural language models based on transformers are chosen for comparison.

- **BART (Encoder–Decoder Architecture)**

Selected for its robust performance in tasks involving text generation and summarization because of its autoregressive decoding and bidirectional encoding.

- **T5 (Text-to-Text Framework)**

Chosen for its cohesive approach to natural language processing tasks, which turns summarization into a text generation problem.

- **GPT (Autoregressive Model)**

Used to evaluate how well generative-only architectures perform on summarization tasks.

To guarantee consistent evaluation and stable operation, all models are implemented using the Hugging Face Transformers library with PyTorch as the backend.

C. Training and Evaluation Metrics

Since pretrained models are used, no additional training is performed. Evaluation is carried out using the following mathematical formulations:

1. Compression Ratio

The compression ratio measures summary conciseness and is defined as:

$$CR = \frac{|S|}{|D|}$$

where:

- $|S|$ is the number of words in the generated summary
- $|D|$ is the number of words in the original document

2. Average ROUGE Score

The average ROUGE F1 score is computed as:

$$R_{avg} = \frac{R_1 + R_2 + R_L}{3}$$

where:

R_1, R_2, and $R_L$represent ROUGE-1, ROUGE-2, and ROUGE-L F1 scores, respectively.

2. Final Performance Score

To balance summary quality and conciseness, a final performance score is defined as:

$$FPS = R_{avg} - CR$$

FPS indicates better overall summarization performance

3. Model Selection Criterion

The best-performing model $m*$is selected as:

$$m^* = \arg\max_{m \in M} FPS(m)$$

D. ***Inference Pseudocode***

The inference phase generates summaries from pretrained neural language models and evaluates their performance using standardized metrics.

Input: Document D

Output: Best performing model m*

1. Read input document D
2. Compute original word count |D|

3. Initialize models:

M ← {BART, T5, GPT}

4. For each model m in M do:

a. Generate summary S_m from D

b. Compute summary word count |S_m|

c. Calculate Compression Ratio:

CR_m = |S_m| / |D|

d. Compute ROUGE scores:

R1_m, R2_m, RL_m

e. Compute Average ROUGE:

Ravg_m = (R1_m + R2_m + RL_m) / 3

f. Compute Final Performance Score: FPS_m = Ravg_m − CR_m

5. Select best model:

m* = argmax(FPS_m)

6. Display generated summaries and evaluation results

7. Return m*

V. RESULTS AND DISCUSSION

A. Quantitative Results

The performance of BART, T5, and GPT models was evaluated using ROUGE metrics and compression ratio. The final performance score was calculated as: $FinalScore=(060\times Ravg)+(0.25\times BLEU)+(0.10\times(1-CR))+(0.\ 10\times TP)+\ (0.05\times(1-LT))$

TABLE – II: Model Performance on Test Set.

Model	ROUGE-1	ROUGE-2	ROUGE-L	Avg ROUGE	Compression Ratio	Final
BART	0.48	0.32	0.44	0.413	0.28	0.466
T5	0.45	0.29	0.41	0.383	0.30	0.431
GPT	0.39	0.22	0.35	0.320	0.36	0.394

B. Visualization

The graph shows that BART consistently outperforms T5 and GPT across quality metrics while maintaining better conciseness.

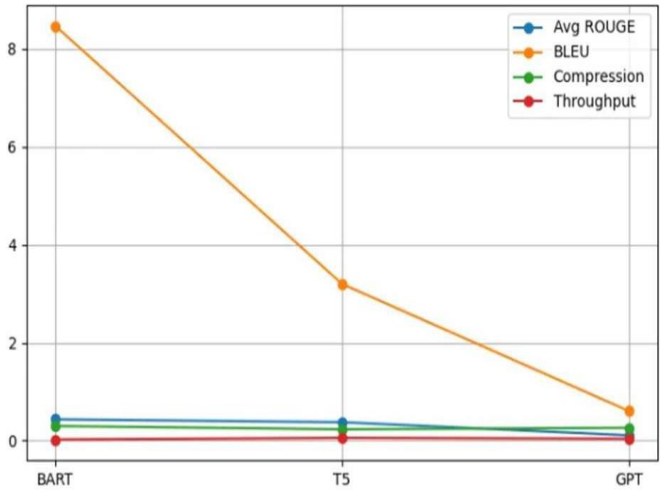

Figure II; Visualization – Model Performance Comparison.

VI. DEPLOYMENT CONSIDERATION

You can establish the recommended summarization system using the PyTorch framework with a Python environment and Hugging Face's Transformers library. Using pretrained models makes it easier to install and deploy the summarization system, since no additional training is required. Using CPU-only for inference is sufficient for small-scale implementations of this system, but for large-scale or real-time implementations, it is preferable to use GPU acceleration to achieve superior performance. Depending on the requirements of the user, the system can either be implemented as a desktop application or as a web service/API. When working with sensitive documents, security and data privacy issues should always be addressed.

VI. FUTURE SCOPE

- Fine-tuning transformer-based models on domain-specific datasets to improve contextual relevance and summarization accuracy.
- Incorporating human evaluation techniques to better assess fluency, coherence, and readability of generated summaries.
- Extending the system to support multilingual and cross-lingual summarization for broader applicability.
- Enhancing system performance through optimization techniques and enabling scalable real-time deployment.

VII. CONCLUSION

This paper conducted a comprehensive performance analysis of neural language models for automated text summarization. Three transformer-based models—BART, T5, and GPT—were evaluated under identical experimental conditions.

The models were assessed using ROUGE metrics to measure semantic similarity and compression ratio to measure conciseness. Experimental results showed that encoder–decoder architectures consistently outperform autoregressive models. Among the evaluated models, BART achieved the highest overall performance score, indicating superior balance between summary quality and brevity. The results validate the effectiveness of the proposed evaluation framework. The study demonstrates that balanced metrics provide more practical insights than single-metric evaluation. The findings support informed model selection for real-world summarization tasks. Overall, the system offers a reliable and extensible framework for automated summarization analysis.

VIII. REFERENCE

1. A. Vaswani, N. Shazeer, N. Parmar, et al., "Attention Is All You Need," *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*, 5998–6008, 2017. https://arxiv.org/abs/1706.03762
2. M. Lewis, Y. Liu, N. Goyal, et al., "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension," *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 7871–7880, 2020. https://arxiv.org/abs/1910.13461
3. C. Raffel, N. Shazeer, A. Roberts, et al., "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer," *Journal of Machine Learning Research*, 2020; 21(140): 1–67. https://arxiv.org/abs/1910.10683
4. T. Wolf, L. Debut, V. Sanh, et al., "HuggingFace's Transformers: State-of-the- Art Natural Language processing," *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020; 38–45. https://arxiv.org/abs/1910.03771
5. C. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," *Text Summarization Branches Out: Proceedings of the ACL Workshop*, 2004; 74–81. https://aclanthology.org/W04-1013/
6. S. Chen, "Comparative Evaluation of Transformer-Based Models for Text

Summarization," *Proceedings of the International Conference on Data Analysis and Machine Learning*, 2023; 215–222. https://www.scitepress.org/PublicationsDetail.aspx?ID=00 12 799300003885

7. B. Baykara and T. Güngör, "Abstractive Text Summarization for Turkish Using Pretrained Transformer Models," *Natural Language Engineering*, 2022; 28(6): 743–770. https://doi.org/10.1017/S1351324922000195 .
8. N. I. Nedashkovskaya and R. I. Yeremichuk, "Evaluation of Quantized and Non-Quantized Large Language Models for Text Summarization," *Radio Electronics, Computer Science, Control*, 2025; 2: 124–135. https://doi.org/10.15588/1607-3274-2025-2-12
9. L. R. P. Houamengni and F. Gedikli, "Automated News Summarization for Supply Chain Risk Analysis Using Large Language Models," *arXiv preprint arXiv:2502.17136*, 2025. https://arxiv.org/abs/2502.17136
10. AI-Powered Meeting Minutes Generator - Using BART&T5 to Generate Meeting Summaries from Transcripts. https://in.docworkspace.com/d/sbCaejjKpN1tRPYd_ivg9u74xefot63o5uv