

---

## Long-Context Transformer Models for Meeting Summarization: A Comparative Study of Full Fine-Tuning and Parameter-Efficient Tuning

Edi Winarko<sup>1</sup>, Katarina Keishanti Joanne Kartakusuma<sup>2</sup>

ewinarko@ugm.ac.id<sup>1</sup>, katarina.kei2003@mail.ugm.ac.id<sup>2</sup>

<sup>1,2</sup>Department of Computer Science and Electronics, Universitas Gadjah Mada, Yogyakarta, Indonesia

---

### Article Information

Received : 30 Nov 2025

Revised : 23 Dec 2025

Accepted : 28 Dec 2025

---

### Keywords

Multi-document summarization, Long-context transformer, Fine-tuning, LoRA, Meeting dataset

---

### Abstract

The growing volume of virtual meetings has increased the need for effective long-document summarization systems that capture essential discussion points from lengthy transcripts. However, existing transformer-based models often struggle to handle long-context inputs and require substantial computational resources for fine-tuning. Moreover, prior work provides limited comparative analysis of full fine-tuning and parameter-efficient fine-tuning (PEFT) specifically for meeting summarization tasks. This study systematically evaluates three long-sequence Transformer architectures—LongT5, BigBird, and LED—on the MeetingBank dataset using both full fine-tuning and PEFT strategies. Models are assessed through ROUGE scores, BERTScore, parameter efficiency, and qualitative error analysis. Experimental results show that LongT5 with full fine-tuning achieves the best performance (ROUGE-1 = 0.675, BERTScore F1 = 0.921), outperforming BigBird as the next-best model by 31.6% in ROUGE-1. PEFT reduces trainable parameters by over 90% and remains competitive only for LongT5 (ROUGE-1 = 0.543, BERTScore F1 = 0.872), while BigBird and LED experience severe degradation, producing semantically weak and incoherent summaries despite low validation loss. These findings highlight that PEFT effectiveness is highly model-dependent and that validation loss alone is an unreliable indicator of generative quality. The study contributes a comprehensive benchmarking analysis and practical insights into optimizing long-document meeting summarization under computational constraints.

## A. Introduction

The widespread adoption of virtual meeting platforms such as Google Meet, Zoom, and Microsoft Teams has fundamentally transformed communication and collaboration in both professional and academic environments. This shift became particularly prominent during the COVID-19 pandemic, as Work From Home (WFH) arrangements increased the volume and frequency of online meetings. By 2023, Microsoft Teams alone reported an extraordinary growth of over 40,000% since 2018, reaching 320 million users [1]. As virtual meetings have become routine, challenges such as "online meeting fatigue" have emerged, where individuals struggle to retain or extract key information from lengthy discussions. Efficiently summarizing essential meeting content is therefore increasingly critical for productivity and information management in organizational settings. Manual review of meeting notes or recordings is time-consuming and impractical for frequent or extended meetings, prompting a strong interest in automated summarization methods.

Automated text summarization techniques are generally categorized into extractive and abstractive approaches. Extractive methods identify and select key sentences directly from the source text, but often fail to maintain coherence or capture the central themes of extended discussions [2]. In contrast, abstractive summarization systems generate concise, coherent summaries by rephrasing and restructuring content, offering greater flexibility and contextual accuracy [3]. Recent advances in abstractive summarization have been driven by large language models (LLMs) such as BART [4], [5], [6], Pegasus [7], [8], and T5 [9], which leverage transformer architectures and large-scale pre-training to produce semantically rich and contextually meaningful summaries [10].

Findings from existing literature highlight the strong capabilities of these LLMs for long-document summarization, including scientific articles and extended text inputs. For instance, Keswani et al. (2024) [3] demonstrated their effectiveness in capturing nuanced long-context structures, while Ülker and Özer (2024) [10] showed their potential to generate domain-specific content through fine-tuning. However, prior research also emphasizes key limitations of default LLM configurations, particularly when applied to meeting summarization. These limitations include difficulty capturing speaker-specific content, maintaining coherent dialogue flow, and avoiding hallucination—where models generate plausible but factually incorrect or irrelevant information [11]. Falke et al. (2019) [12], for example, reported that approximately 25% of summaries produced by state-of-the-art systems contained hallucinated content.

Meeting summarization presents unique challenges that differ from general text summarization. Laskar et al. (2023) [13] highlight the complexity of condensing multi-speaker discussions while preserving contextual nuances and conversational flow. Similarly, Lodhi et al. (2022) [14] emphasize the need for meeting summaries to balance brevity and completeness, ensuring that essential information is included without overwhelming the reader. These considerations underscore the necessity of fine-tuning LLMs on domain-specific datasets such as MeetingBank to ensure relevance, accuracy, and coherence in generated summaries. Studies such as Dang et al. (2022) [15] demonstrate that targeted fine-tuning strategies—e.g., topic

modeling integration—can significantly improve the quality of abstractive summaries.

Despite advancements in abstractive summarization, existing research reveals a significant gap in the systematic evaluation of large language models for meeting-specific summarization tasks. While models like BART, Pegasus, and T5 exhibit strong performance on general summarization tasks, their effectiveness diminishes when applied to meeting transcripts due to challenges such as capturing speaker-specific dialogue, preserving discussion flow, and managing context-dependent nuances [13], [14]. Furthermore, models pre-trained on diverse general-purpose corpora frequently exhibit hallucination or produce irrelevant content when used for domain-specific summarization tasks [3].

To address these limitations, this research investigates the fine-tuning of long-sequence Transformer architectures—specifically BART, Pegasus, and T5-based models—on the MeetingBank dataset, a resource tailored to the structural and conversational characteristics of meeting transcripts. By optimizing hyperparameters and evaluating both full fine-tuning and parameter-efficient fine-tuning (PEFT) strategies, this study aims to enhance the contextual accuracy, coherence, and brevity of meeting summaries. The findings contribute insights into effective fine-tuning configurations and model selection strategies, addressing a critical gap in automated meeting summarization and advancing its applicability in real-world professional and academic contexts.

## B. Research Method

This study investigates the effectiveness of long-context transformer models for meeting summarization by fine-tuning three pretrained encoder-decoder architectures—LongT5 [9], BigBird-Pegasus [7], and LED [16]—on the MeetingBank dataset. The overall workflow consists of dataset preparation and filtering, data cleaning and preprocessing, model fine-tuning under full and parameter-efficient regimes, and intrinsic evaluation using lexical and semantic metrics (see Figure 1 for the proposed workflow). All experiments are implemented in Python using the Hugging Face Transformers ecosystem and executed in Jupyter Notebook on an NVIDIA A100 GPU.

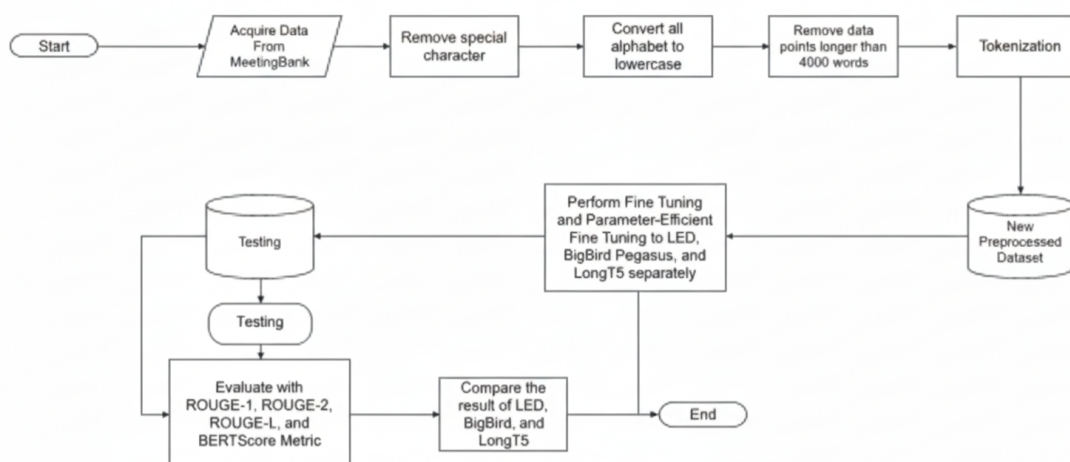
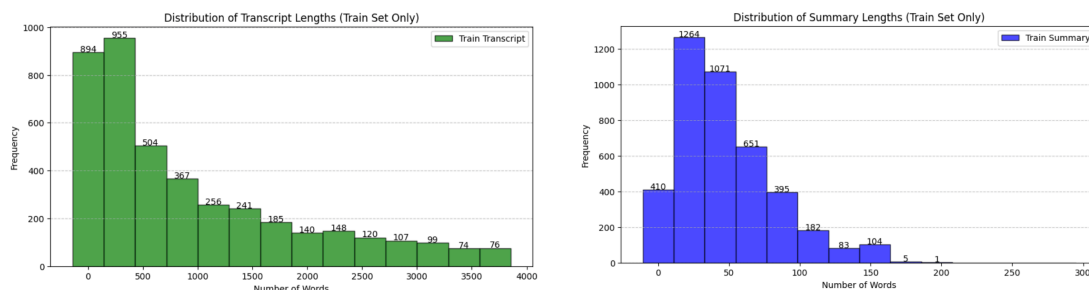


Figure 1. Research workflow

## Dataset and Exploratory Analysis

The experiments use the MeetingBank dataset, a meeting summarization corpus containing transcripts and abstractive summaries. The original split comprises 5,169 training, 862 testing, and 861 validation instances, with four main fields: *transcript (model input)*, *summary (reference label)*, *uid*, and *id*. Because the models under study support a maximum context length of 4,096 tokens, the dataset is first filtered to discard transcripts exceeding this limit to avoid truncation of crucial content. After filtering, 4,166 samples are retained for training, 710 for testing, and 716 for validation.

The two histograms in Figure 2 illustrate the length distribution of transcripts and summaries in the filtered training set, revealing a substantial mismatch between the verbosity of meeting transcripts and the conciseness of their corresponding summaries. As shown in Figure 2(a), transcript lengths exhibit a long-tailed distribution, with most transcripts falling below 600 words, yet a significant number extending beyond 1,500 words and some approaching 4,000 words. This wide variability highlights the inherent complexity of meeting conversations, which often contain extended dialogue, topic shifts, and speaker turns. In contrast, Figure 2(b) shows that summaries are far more compact, with the majority concentrated between 20 and 100 words and only a small fraction exceeding 150 words. The stark difference between the two distributions underscores the challenging compression ratio required for abstractive meeting summarization and indicates that models must not only condense large amounts of information but also maintain coherence and relevance when generating much shorter outputs.



(a) Transcript length distribution

(b) Summary length distribution

Figure 2. Length distribution of transcripts and summaries in the training set after filtering

Additional exploratory analysis examines the top-20 most frequent tokens in transcripts and summaries shows that transcripts are dominated by conversational function words (e.g., “you”, “I”, “we”), whereas summaries concentrate more on domain-specific terms (e.g., “city”, “ordinance”, “council”), reflecting their more formal and condensed nature.

## Data Cleaning and Preprocessing

A light text-cleaning stage is applied to standardize the input and reduce noise. This includes removing non-alphanumeric special characters and lowercasing all text to ensure consistency during tokenization. No aggressive normalization (e.g.,

stemming or lemmatization) is performed, as modern transformer tokenizers are robust to surface variation.

Preprocessing is then tailored to the tokenizer associated with each model. LongT5 and BigBird-Pegasus employ SentencePiece tokenization, which keeps frequent words intact but decomposes rarer words into subwords, enabling efficient handling of long sequences while preserving semantics. LED uses a byte-level BPE tokenizer that splits text into smaller subword units and marks word boundaries with a special “Ġ” prefix. For all models, input and output sequences are padded up to the maximum sequence length of 4,096 tokens. Padding tokens are ignored by the attention mechanism but ensure that all sequences in a batch share the same length; LongT5 and BigBird-Pegasus use “0” as the padding index, whereas LED uses “1”.

### **Fine-Tuning Strategy and Hyperparameters**

The core of the methodology is the adaptation of LongT5, BigBird-Pegasus, and LED to the meeting summarization task through two regimes: (i) full fine-tuning, where all model parameters are updated, and (ii) parameter-efficient fine-tuning (PEFT), where only a small set of additional parameters is trained while the base model is kept frozen.

For full fine-tuning, pretrained checkpoints are initialized with their default weights and optimized using the cross-entropy loss between generated and reference summaries. Training is conducted for 3, 4, and 5 epochs to explore convergence behavior, with validation loss monitored for early stopping and overfitting indications. The optimizer is AdamW with a learning rate of  $3 \times 10^{-5}$ , consistent with standard practices in fine-tuning large language models. Because of the substantial memory footprint of 4,096-token sequences, batch sizes are restricted to 1–2 examples per step; gradient accumulation is employed to simulate larger effective batch sizes without exceeding GPU memory constraints.

For PEFT, the study employs Low-Rank Adaptation (LoRA) modules integrated into each model's attention layers. In this setup, the original transformer weights remain frozen, and only low-rank adaptation matrices are trained, drastically reducing the number of trainable parameters and computational cost. The LoRA configuration uses rank  $r = 32$  and  $\alpha = 64$  (i.e.,  $\alpha = 2r$ ), a setting reported in prior work to provide a good balance between efficiency and performance. LoRA-enhanced models are trained under the same data splits, sequence length, and optimization settings as the full fine-tuning models to allow a fair comparison between regimes.

### **Evaluation Metrics**

Model performance is evaluated on the held-out test set using both lexical overlap and semantic similarity metrics. For lexical evaluation, the study employs ROUGE-1, ROUGE-2, and ROUGE-L. ROUGE-1 measures unigram overlap between generated and reference summaries, capturing overall content preservation; ROUGE-2 measures bigram overlap, reflecting phrase-level coherence; and ROUGE-L computes the longest common subsequence, assessing sentence-level ordering and fluency. ROUGE is particularly suitable for meeting summarization because recall-oriented coverage of salient content is crucial: summaries must include all key decisions and points discussed, even at the cost of some redundancy [17].

To complement ROUGE and address its limitations in capturing semantic equivalence, the study additionally reports BERTScore, which uses contextual embeddings from a pretrained language model to quantify semantic similarity between generated and reference summaries [17]. BERTScore provides a more nuanced view of meaning preservation, particularly in cases where paraphrasing leads to low n-gram overlap but high semantic fidelity. By combining ROUGE and BERTScore, this research aims to present a comprehensive evaluation of the fine-tuned models' ability to handle the complexities of meeting summarization effectively.

### C. Result and Discussion

The experimental evaluation investigates the performance of three long-context transformer architectures—LongT5, BigBird-Pegasus, and LED—fine-tuned on the MeetingBank dataset using both full fine-tuning and parameter-efficient fine-tuning (PEFT) with LoRA. The analysis integrates training dynamics, intrinsic evaluation using ROUGE and BERTScore, and output length behavior.

#### Training Stability and Convergence Analysis

Across all configurations, the training and validation loss curves reveal apparent differences in optimization stability among models. As shown in Figure 3, LongT5 demonstrates consistently smooth convergence under both full fine-tuning and PEFT. In the full fine-tuning setting, LongT5 reaches a best validation loss of 0.4391, while the PEFT variant attains 0.4898 with similarly stable trends. The relatively small gap between training and validation loss indicates strong generalization and minimal overfitting.

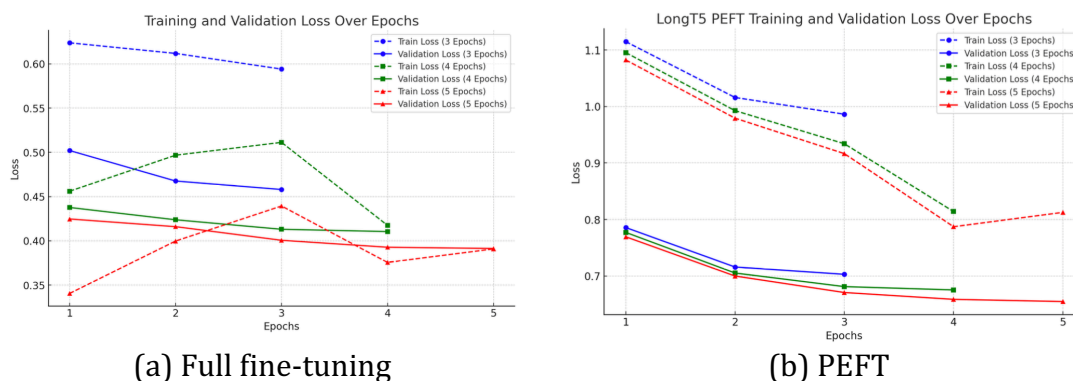
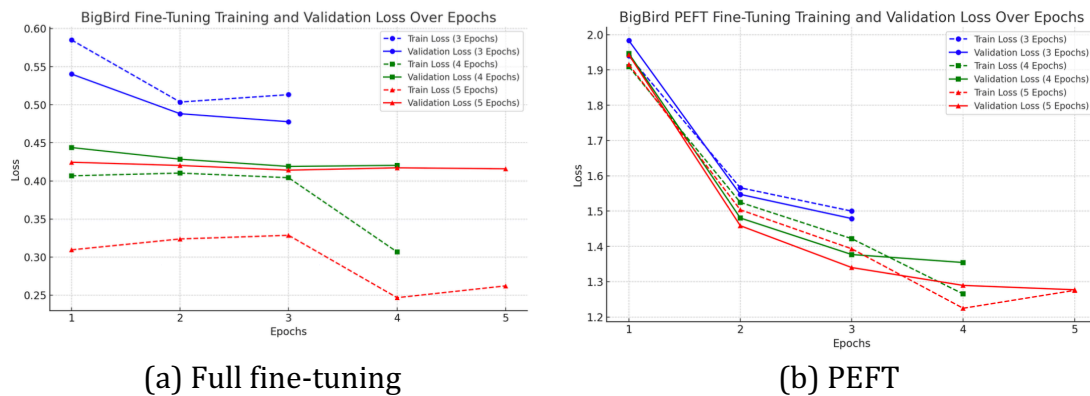
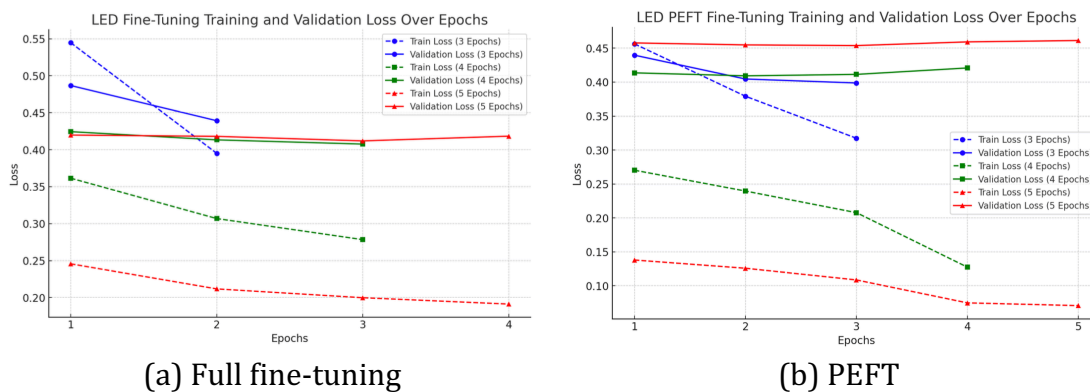


Figure 3. Training and validation loss of LongT5: (a) full fine-tuning, (b) PEFT

BigBird-Pegasus, shown in Figure 4, exhibits stable convergence under full fine-tuning with a best validation loss of 0.4141. However, when trained using PEFT, BigBird shows substantial instability, with validation loss rising to 1.6043 and fluctuating heavily across steps—suggesting that LoRA is insufficient to adapt BigBird's sparse-attention structure for long-document summarization. LED displays an even more striking discrepancy between numerical and practical performance. Figure 5 shows that the LED achieves the lowest validation losses across the entire experiment (0.4043 for full fine-tuning and 0.3987 for PEFT).



(a) Full fine-tuning (b) PEFT  
 Figure 4. Training and validation loss of BigBird: (a) full fine-tuning, (b) PEFT



(a) Full fine-tuning (b) PEFT  
 Figure 5. Training and validation loss of LED: (a) full fine-tuning, (b) PEFT

**Quantitative Evaluation on Test Set**

A quantitative comparison of all models on the test set is presented in Tables 1 and 2. The results show that LongT5 with full fine-tuning outperforms all other variants by a large margin, achieving ROUGE-1 of 0.675, ROUGE-2 of 0.585, ROUGE-L of 0.659, and BERTScore F1 of 0.921. BigBird-Pegasus under full fine-tuning performs moderately well but trails LongT5 substantially, particularly in ROUGE-2 and ROUGE-L, indicating reduced phrase-level and structural coherence. LED full fine-tuning achieves reasonable unigram overlap but extremely low ROUGE-2, suggesting a lack of multi-word coherence in its summaries. Under PEFT, performance degrades sharply for most models. LongT5 PEFT remains the only configuration that preserves good performance, achieving ROUGE-1 of 0.543 and BERTScore F1 of 0.895. In contrast, BigBird-Pegasus PEFT and LED PEFT show severe performance drops, with ROUGE-1 scores of 0.115 and 0.075, respectively. These results confirm that full fine-tuning is essential for high-quality long-document summarization, and that LoRA-based PEFT is only viable for architectures whose attention mechanisms can adapt to low-rank updates—namely LongT5.

Table 1. ROUGE score comparison of all models on the test set

Type	Model	ROUGE-1	ROUGE-2	ROUGE-L
Fine Tune	LongT5	<b>0.675</b>	<b>0.585</b>	<b>0.659</b>

	BigBird	0.330	0.255	0.313
	LED	0.170	0.050	0.138
PEFT	LongT5	0.543	0.439	0.523
	BigBird	0.115	0.009	0.093
	LED	0.075	0.008	0.063

Table 2. BERTScore comparison of all models on the test set

Type	Model	BERTScore Precision	BERTScore Recall	BERTScore F1
Fine Tune	LongT5	<b>0.934</b>	<b>0.909</b>	<b>0.921</b>
	BigBird	0.732	0.712	0.721
	LED	0.776	0.811	0.792
PEFT	LongT5	0.909	0.882	0.895
	BigBird	0.744	0.783	0.763
	LED	0.301	0.293	0.302

### Analysis of Summary Length and Output Characteristics

Further insight is provided by examining the distribution of generated summary lengths in Figures 6, 7, and 8. In Figure 6, LongT5 full fine-tuning produces shorter but more information-dense summaries, suggesting efficient content selection and reduced redundancy. LongT5 PEFT generates moderately more extended summaries, reflecting less compression but maintaining strong semantic fidelity. In contrast, BigBird-Pegasus PEFT frequently produces abnormally long summaries characterized by repetition and topic drift (Figure 7). LED, in both fine-tuning modes, suffers from excessive repetition and looping structures, often generating verbose segments that do not correspond to the input content (Figure 8).

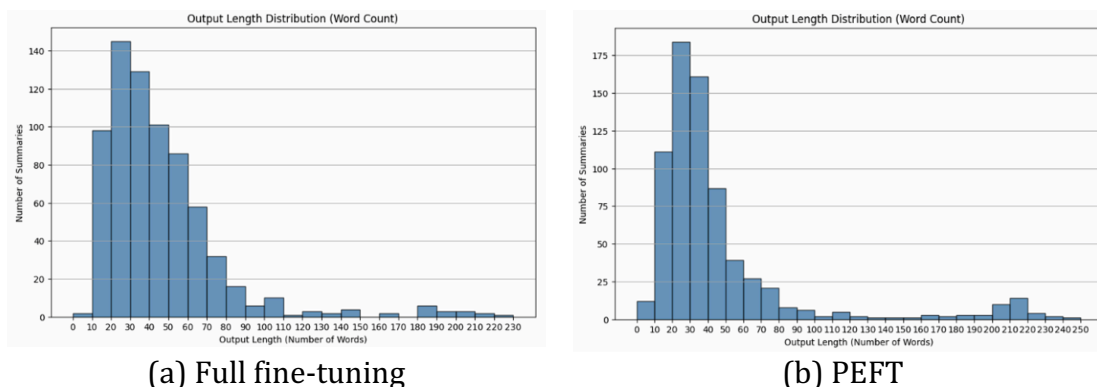
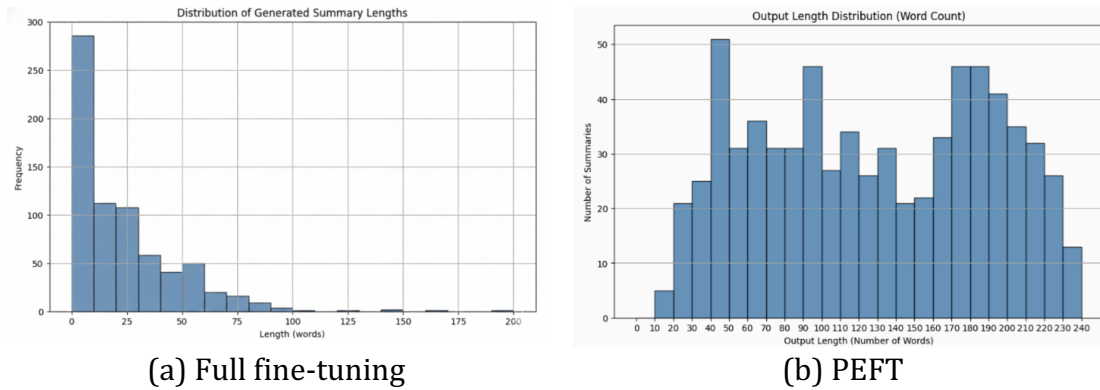
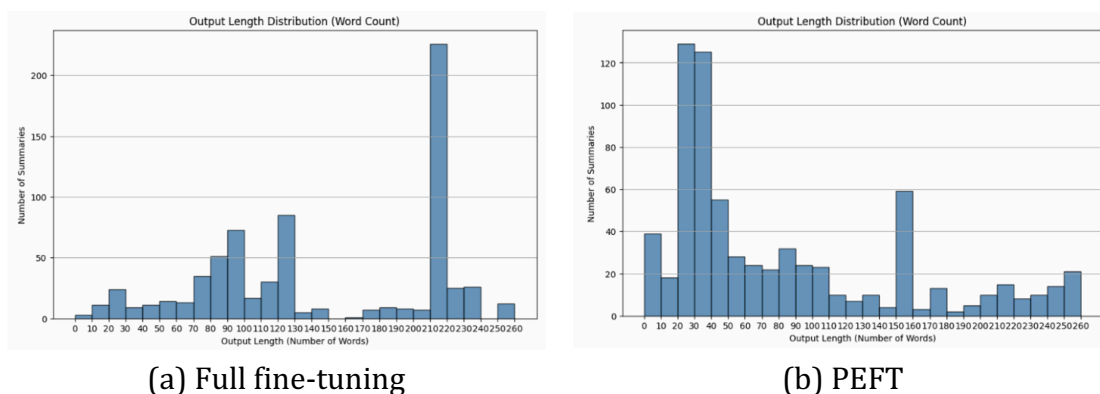


Figure 6. Generated summary length distribution of LongT5: (a) full fine-tuning, (b) PEFT



(a) Full fine-tuning (b) PEFT  
 Figure 7. Generated summary length distribution of BigBird: (a) full fine-tuning, (b) PEFT



(a) Full fine-tuning (b) PEFT  
 Figure 8. Generated summary length distribution of LED: (a) full fine-tuning, (b) PEFT

These findings highlight several important insights. First, LongT5 with full fine-tuning is the most robust and effective architecture for long-document meeting summarization, outperforming all other models by a substantial margin in both linguistic and semantic metrics. Second, PEFT is highly architecture-dependent: LongT5 can successfully leverage LoRA, but BigBird-Pegasus and LED cannot, due to their underlying attention mechanisms and representational constraints. Third, validation loss alone is insufficient to estimate generation quality, particularly for LED, which converges numerically yet fails to produce coherent summaries. Finally, hallucination, redundancy, and attention instability remain prominent challenges in long-document abstractive summarization models, especially under parameter-efficient regimes.

Overall, the results demonstrate that full fine-tuning is necessary for achieving high-quality meeting summarization performance in long-context transformer architectures. At the same time, parameter-efficient methods must be carefully matched to the model architecture. LongT5 stands out as the most reliable and adaptable model across all evaluation dimensions.

#### D. Conclusion and Future Work

This study evaluated three long-context transformer models—LongT5, BigBird-Pegasus, and LED—for abstractive meeting summarization under full fine-tuning and parameter-efficient LoRA tuning. The results demonstrate that LongT5 is the

most reliable and effective model, consistently producing coherent and semantically faithful summaries. It also proved to be the only architecture that maintained competitive performance under LoRA, indicating stronger compatibility with parameter-efficient adaptation.

In contrast, BigBird-Pegasus and LED showed significant limitations, especially under PEFT, including instability, repetition, and hallucinations. These findings highlight that model architecture plays a crucial role in the suitability of PEFT techniques for long-document summarization. The study underscores that, for complex multi-speaker meeting transcripts, full fine-tuning remains the most dependable approach, while PEFT requires careful architectural alignment.

Several directions can extend the present work. Future studies may investigate more advanced parameter-efficient methods, such as AdaLoRA, prefix tuning, or hybrid strategies, to improve adaptation in architectures that performed poorly under LoRA. Improving evaluation frameworks—for example, by incorporating factual consistency metrics, hallucination detection, or summary-level semantic scoring—could provide more reliable signals during training and better capture generative quality. Exploring additional summarization datasets or multi-domain meeting corpora would help assess generalizability beyond MeetingBank.

## E. Acknowledgment

This work is supported in part by the Department of Computer Science and Electronics, Faculty of Mathematics and Natural Sciences, UGM, Schema B Research Grant No. 4138/UN1/FMIPA.1.3/KP/PT.01.03/2025.

## F. References

- [1] R. Shewale, "Microsoft Teams Statistics - Users & Revenue (2023 report)," 2023, *demandsage*. [Online]. Available: <https://www.demandsage.com/microsoft-teams-statistics/>
- [2] M. Majeed and K. M. T, "Comparative study on extractive summarization using sentence ranking algorithm and text ranking algorithm," in *2023 International Conference on Power, Instrumentation, Control and Computing (PICC)*, 2023. doi: 10.1109/picc57976.2023.10142314.
- [3] G. Keswani, W. Bisen, H. Padwad, Y. Wankhedkar, S. Pandey, and A. Soni, "Abstractive Long Text Summarization using Large Language Models," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 12, no. 12s, pp. 160–168, 2024, [Online]. Available: <https://ijisae.org/index.php/IJISAE/article/view/4500>
- [4] M. Lewis *et al.*, "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 7871–7880.
- [5] M. La Quatra and L. Cagliero, "BART-IT: An Efficient Sequence-to-Sequence Model for Italian Text Summarization," *Future Internet*, vol. 15, no. 1, p. 15, 2023, doi: 10.3390/fi15010015.
- [6] T. G. Altundogan, M. Karakose, and O. Tokel, "Bart Fine tuning based abstractive summarization of patients medical questions texts," in *2023 4th*

- International Conference on Data Analytics for Business and Industry (ICDABI)*, 2023, pp. 174–178. doi: 10.1109/icdabi60145.2023.10629497.
- [7] J. Zhang, Y. Zhao, M. Saleh, and P. J. Liu, “PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization,” in *Proceedings of the 37th International Conference on Machine Learning*, 2020, pp. 11328–11339.
- [8] N. Nalini, A. Narayan, A. M. Sridharan, and A. Pradhan, “Automated Text Summarizer Using Google Pegasus,” in *2023 International Conference on Smart Systems and Inventive Technology (ICSSIT)*, IEEE, 2023. doi: 10.1109/ICSSIT48917.2023.1019972.
- [9] M. Guo, J. Lee, Y. I. Chiu, S. Narang, S. Ruder, and C. Raffel, “LongT5: Efficient Text-to-Text Transformer for Long Sequences,” *arXiv preprint arXiv:2203.05502*, 2022.
- [10] M. Ülker and A. B. Özer, “Abstractive Summarization Model for Summarizing Scientific Article,” 2024, doi: 10.20944/preprints202405.1123.v1.
- [11] Z. Ji, N. Lee, J. Fries, T. Yu, and C. Finn, “Survey of Hallucination in Natural Language Generation,” 2023, doi: <https://doi.org/10.1145/3571730>.
- [12] T. Falke, L. F. R. Ribeiro, P. A. Utama, I. Dagan, and I. Gurevych, “Ranking generated summaries by correctness: an interesting but challenging application for natural language inference,” *Ranking Generated Summaries by Correctness: An Interesting but Challenging Application for Natural Language Inference*, 2019, doi: 10.18653/v1/p19-1213.
- [13] M. T. R. Laskar and et al., “Building real-world meeting summarization systems,” 2023, [Online]. Available: <https://aclanthology.org/2023.emnlp-industry.33.pdf>
- [14] P. M. Lodhi, S. Kharche, D. Kambri, and S. S. Khan, “Business Meeting Summarization System,” in *2022 2nd Asian Conference on Innovation in Technology (ASIANCON)*, Ravet, India: IEEE, 2022, pp. 1–6. doi: 10.1109/ASIANCON55314.2022.9908905.
- [15] B. Dang, D.-T. Do, and L.-M. Nguyen, “TBART: Abstractive summarization based on the joining of topic modeling and Bart,” in *2022 14th International Conference on Knowledge and Systems Engineering (KSE)*, 2022, pp. 1–6. doi: 10.1109/kse56063.2022.9953613.
- [16] I. Beltagy, M. E. Peters, and A. Cohan, “Longformer: The long-document transformer,” *arXiv preprint arXiv:2004.05150*, 2020.
- [17] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, “BERTScore: Evaluating text generation with BERT,” in *Proceedings of the 8th International Conference on Learning Representations (ICLR)*, 2020, pp. 1–12.