Improving the Efficiency of Long Document Classification using Sentence Ranking Approach

Prathamesh Kokate[1,3], Mitali Sarnaik[1,3], Manavi Khopade[1,3], and Raviraj Joshi[2,3]
[1]Pune Institute of Computer Technology, Pune
[2]Indian Institute of Technology Madras, Chennai
[3]L3Cube Labs, Pune

Abstract

Long document classification poses challenges due to the computational limitations of transformer-based models, particularly BERT, which are constrained by fixed input lengths and quadratic attention complexity. Moreover, using the full document for classification is often redundant, as only a subset of sentences typically carries the necessary information. To address this, we propose a TF-IDF-based sentence ranking method that improves efficiency by selecting the most informative content. Our approach explores fixed-count and percentage-based sentence selection, along with an enhanced scoring strategy combining normalized TF-IDF scores and sentence length. Evaluated on the MahaNews Long Document Classification (LDC) dataset of long Marathi news articles, the method consistently outperforms baselines such as first, last, and random sentence selection. With MahaBERT-v2, we achieve near-identical classification accuracy with just a 0.33 percent drop compared to the full-context baseline, while reducing input size by over 50 percent and inference latency by 43 percent. This demonstrates that significant context reduction is possible without sacrificing performance, making the method practical for real-world long document classification tasks.

1 Introduction

Long document classification is vital in NLP applications such as research, legal, news, and reviews. Transformer models like BERT achieve strong results but are limited by input size and high attention costs (Park et al., 2022; Zaheer et al., 2020), often requiring truncation or complex hierarchical processing (Wagh et al., 2021; Devlin et al., 2018). We propose a data-driven approach (Minaee et al., 2021) that ranks and selects key sentences using TF-IDF (Qaiser and Ali, 2018), treating each sentence as a document and summing term scores (Das and Chakraborty, 2020; Kim and Gil, 2019; Liu et al., 2018a; Das et al., 2023). High-ranking sentences capture domain-specific context while minimizing input length, enabling efficient classification with reduced computational overhead (Figure 1).
We explore multiple strategies for selecting sentences, including the following:

1. Fixed-length selection involves choosing a predefined number of top-ranked sentences, with evaluations conducted for 1, 2, 3, 4, and 5 sentences.

2. Percentage-based selection refers to the selection of a specific percentage of top-ranked sentences, varying from 10% to 100% in increments of 10%.

3. Weighted ranking combines normalized TF-IDF scores with sentence length to balance importance and informativeness, exploring different weighting factors to identify the optimal configuration.

To evaluate the effectiveness of these strategies, we conduct extensive experiments on the MahaNews[1] dataset (Mittal et al., 2023; Aishwarya et al., 2023), a corpus of long Marathi news articles categorized by topic. Using MahaBERT[2](marathi-bert-v2) (Joshi, 2022), we train and test models on reduced-context versions of the dataset and compare the classification performance across different selection methods. Our results demonstrate that TF-IDF-based ranking significantly outperforms simpler selection strategies, such as choosing the first, last, or randomly sampled sentences. Additionally, integrating length-aware weighting further enhances accuracy, while context reduction leads

[1]https://github.com/l3cube-pune/indic-nlp/tree/refs/heads/main/L3Cube-IndicNews/Marathi/LDC
[2]https://huggingface.co/l3cube-pune/marathi-bert-v2

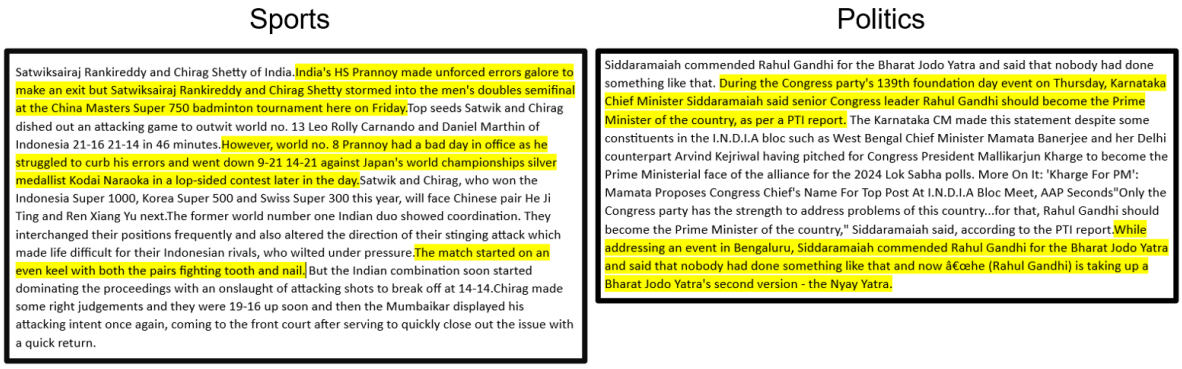

Figure 1: Illustration of key idea — selective sentence processing for efficient document classification. The figure presents two example paragraphs, representing only a portion of the long documents: one related to sports and the other to politics. In each case, the most semantically relevant and contextually informative sentences are highlighted. These highlighted sentences contain domain-specific cues (e.g., sports activities or political entities) that enable accurate classification without processing the full document. This demonstrates that selective sentence extraction can preserve classification performance while reducing computational overhead.

to a substantial decrease in inference time without compromising performance.

1.1 Key Contributions

- We propose a novel TF-IDF-based sentence ranking and context reduction strategy to improve the efficiency of BERT models for long document classification without altering the model architecture, significantly reducing processing time for large text inputs.
- We evaluate multiple sentence selection techniques such as fixed-length, percentage-based, and weighted ranking, analyzing their trade-offs in balancing efficiency and classification accuracy.
- Experiments on the MahaNews dataset show that ranked selection consistently outperforms naive approaches while maintaining accuracy and significantly reducing inference time. Specifically, the performance of selection strategies follows the order: ranked > first > random > last. Notably, selecting sentences from the beginning of the document serves as a strong baseline.
- Our findings reveal an optimal balance between input length, accuracy, and computational efficiency, demonstrating that selecting a subset of ranked sentences can achieve near-full-document classification performance.

By systematically analyzing context reduction techniques, our work provides a practical and efficient alternative to architectural modifications for long document classification in transformer-based models.

2 Related Work

Long documents contain extensive information, making direct processing with traditional classification models computationally expensive and time-consuming. To improve efficiency, existing methods generally fall into two categories: data-based approaches and model-based approaches.

2.1 Model Based Approaches

Handling long document classification efficiently requires balancing model complexity with computational feasibility. Model-based techniques addressing this challenge include sparse attention mechanisms, quantization, recurrent architectures, and normalization strategies. Sparse attention mechanisms enable transformer models to process significantly longer inputs while retaining the advantages of full-attention models (Pham and The, 2024). By incorporating global tokens for capturing overall context, local tokens for nearby interactions, and random tokens to enhance global coverage, these mechanisms effectively reduce memory and computation costs from quadratic to linear (Martins et al., 2020), making them particularly useful for handling extensive input sequences.
Beyond attention mechanisms, reducing computational demand can be achieved through quantization, which lowers the precision of model weights to save memory. For example, Q8 BERT employs

8-bit weights instead of the standard 32-bit, using techniques such as quantization-aware training (Zafrir, 2019). This approach significantly reduces model size while maintaining accuracy, making it suitable for deployment in resource-constrained environments. Recurrent architectures like Long Short-Term Memory (LSTM) networks have also been explored for capturing long-term dependencies (Teragawa et al., 2021; Putri and Setiawan, 2023). While LSTMs excel at preserving sequential information, their sequential nature limits parallelization, giving transformer-based models an advantage in scalability.

To further improve the stability and efficiency of transformer models, pre-layer normalization is applied. This technique normalizes activations before the attention mechanism, mitigating gradient instability and accelerating convergence (Beltagy et al., 2020). By improving training dynamics, pre-layer normalization enhances the robustness of deep transformer architectures, making them more suitable for long document classification. Combining sparse attention for efficiency, quantization for reduced computational demand, recurrent mechanisms for sequence retention, and pre-layer normalization for stability enables modern NLP models to effectively process long documents while optimizing performance and resource utilization (Al-Qurishi, 2022).

2.2 Data Based Approaches

Unlike model-based approaches that improve architectures and algorithms, data-centric methods optimize the training and testing data pipeline to boost performance without altering the model. For example, Discriminative Active Learning (DAL) reduces labeling effort by selecting informative instances near the decision boundary, ensuring labeled and unlabeled data distributions align in the learned space (Bamman and Smith, 2013). Another strategy tackles transformer input limits by splitting long documents, processing chunks individually, and aggregating results via hierarchical models (Yang et al., 2016; Khandve et al., 2022) or hierarchical attention (Yang et al., 2016). We adopt a data-centric approach for its easy integration, domain-agnostic nature, robustness against model biases, and scalability (Song, 2024; Moro, 2023). By minimizing contextual information during training and inference (He, 2019; Liu et al., 2018b; Tay et al., 2021) through selective input curation, our method adapts across domains and models without architectural changes (Li et al., 2018; Prabhu et al., 2021; Sun et al., 2020).

3 Methodology

The datasets utilized in our experiments are sourced from L3Cube's IndicNews corpus a multilingual text classification dataset curated for Indian regional languages. The MahaNews corresponds to the Marathi subset of the IndicNews dataset. The corpus covers news headlines and articles in 11 prominent Indic languages, with each language dataset encompassing 10 or more news categories. We have made use of the LDC dataset which consists of full articles with their categories (Mittal et al., 2023; Aishwarya et al., 2023).

Our methodology focuses on optimizing input size while preserving classification performance using the Marathi LDC dataset, which consists of full-length articles in Marathi (Jain et al., 2020). We begin by tokenizing each article into individual sentences, followed by computing the TF-IDF score for each sentence. The sentences are then ranked based on their scores, and the context is reduced by selecting top-ranked sentences. To achieve this, we explore various sentence selection strategies. Instead of using the entire article, the selected sentences are fed into the MahaBERT model for classification.

3.1 Training and Testing

We aim to improve classification efficiency by reducing input text during training and inference while maintaining performance comparable to full-document processing. On the full LDC dataset, the MahaBERT model, fine-tuned on L3Cube-MahaCorpus and other public Marathi datasets achieved 94.706% accuracy. Our objective is to approach this accuracy using reduced context inputs. The Marathi LDC dataset contains 20,425 training samples, 2,550 testing samples, and 2,548 validation samples used to enhance model accuracy.

Sentence Selection Techniques

We evaluated several sentence selection strategies, ranging from simple selection methods to a novel TF-IDF-based method. These approaches aim to retain the most informative parts of each document which are used to train the model and subsequently test it.

- **First Few Sentences Selection:** In this

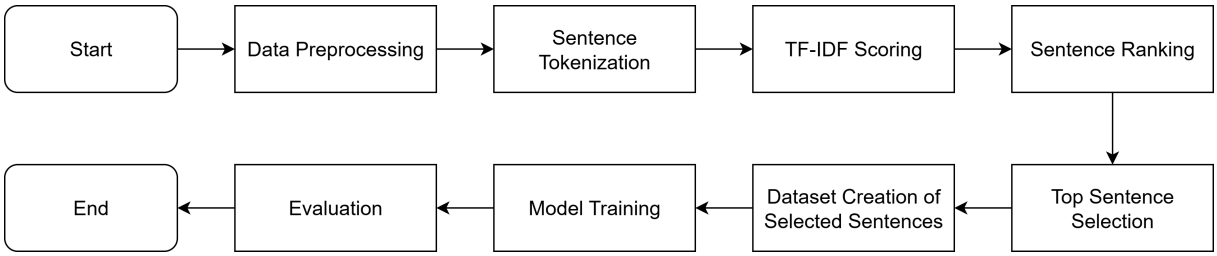

Figure 2: Workflow of ranked approach for sentence selection — The diagram illustrates a ranked sentence selection workflow starting from raw text input. Data is preprocessed and split into sentences, which are scored using TF-IDF. Top-ranked sentences are selected to create a training dataset. This dataset is used for model training and evaluation before concluding the process.

method, only a specified percentage of the initial sentences from each article is selected. This leverages the observation that the opening sentences often contain summaries or key contextual information critical for classification.

- **Last Few Sentences Selection:** Conversely, this method selects only the last portion of sentences from each article. The rationale is that concluding sentences often include detailed analysis or summaries, which may also be useful for accurate classification.

- **Random Sentences Selection:** Here, sentences are randomly selected from across the article. While this approach is computationally efficient and allows for diverse content selection, it is unreliable, as critical information may be excluded, leading to inconsistent classification performance across samples.

While these methods are straightforward and easy to implement, they can fail to consistently capture the document's most relevant content, as important sentences may appear in various parts of the text.

3.2 TF-IDF-Based Ranking and Selection

Random sentence selection, though efficient, ignores sentence importance based on factors like distinctiveness and semantic relevance, leading to inconsistent accuracy in long document classification. We address this with a novel TF-IDF based sentence selection method that ranks sentences by informative value, reducing training and inference time while maintaining high accuracy.

General Flow of the Method

The sentences of each article are preserved in their original order and tokenized using a language-specific tokenizer, such as the Indic NLP tokenizer for the IndicNews dataset.

For TF-IDF score calculation, each sentence is treated as an individual document in the context of determining Term Frequency (TF) and Inverse Document Frequency (IDF). This approach identifies terms that occur frequently within a sentence but are rare across others in the same article, thereby quantifying the importance of each term.

The TF-IDF scores computed for each sentence result in an ordered array where the most informative sentences appear at the top.

Score Computation

The score of a sentence S_i can be calculated as the sum of the TF-IDF scores of all terms t_j within the sentence.

Formally, the score $\text{Score}(S_i)$ is defined as:

$$\text{Score}(S_i) = \sum_{t_j \in S_i} \text{TF-IDF}(t_i)$$

Where,

$$\text{TF-IDF}(t_i) = \text{TF}(t_i) \cdot \text{IDF}(t_i)$$

Definitions

1. **Term Frequency (TF):**

$$\text{TF}(t_j) = \frac{\text{Frequency of } t_j \text{ in } S_i}{\text{Total number of terms in } S_i}$$

2. **Inverse Document Frequency (IDF):**

$$\text{IDF}(t_j) = \log\left(\frac{N}{1 + \text{Sentence frequency of } t_j}\right)$$

where N is the total number of sentences in the article, and the document frequency is the number

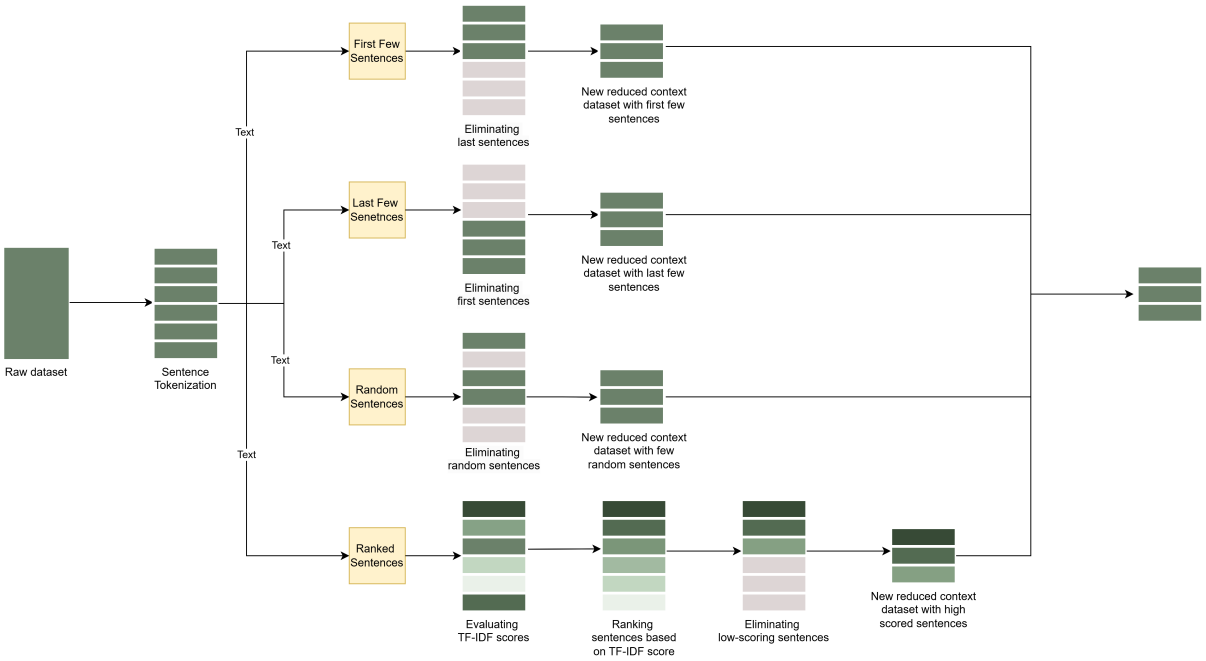

Figure 3: Sentence selection approaches — The image illustrates various sentence selection approaches used for context reduction. Methods include selecting the first few sentences, last few sentences, random sentences, and ranked sentences. In the ranked approach, sentences are scored using TF-IDF and selected based on informative context.

of sentences containing t_j.
This formula ensures that the importance of each sentence is derived from the significance of its terms within the context of the article.
The above approaches, to select sentences, for model are as depicted in Figure 3.

Optimal Number Sentence Selection
Selecting the optimal number of sentences involves balancing efficiency and classification accuracy. Several approaches are considered for determining the most informative subset of sentences:

- **Top-ranked sentence selection:** The highest-ranked sentence, based on its TF-IDF score, is used to evaluate the effectiveness of minimal context in classification.
- **Incremental context expansion:** The top two, three, four, and five ranked sentences are examined to assess the impact of increasing contextual information on classification accuracy and to identify the point of diminishing returns.
- **Percentage-based selection:** Top-ranked sentences are progressively selected in increments of 10%, aiming to find an optimal balance between efficiency and performance. This method is particularly effective for documents.

These approaches help refine sentence selection strategies to enhance both computational efficiency and model performance while minimizing unnecessary information.

3.3 Length Normalization

Normalization scales features to a common range, ensuring fair contribution and preventing dominance due to scale differences. In our case, it adjusts sentence TF-IDF scores to avoid bias toward longer sentences, which would otherwise rank higher simply due to more terms rather than higher informative content.
To ensure fair sentence ranking, different approaches balance sentence length and TF-IDF scores:

- **Length Normalization:** Divides the total TF-IDF score by sentence length to prevent longer sentences from being unfairly ranked higher.
- **Weighted Balancing:** Uses a dynamic weighted formula to balance TF-IDF score and sentence length.

Length normalization (dividing total TF-IDF by token count) ranks sentences by average term im-

portance, allowing fair comparison across different lengths. However, after analyzing the selected sentences it was observed that normalization introduced an inverse bias toward shorter sentences. To address this, we introduce a weighted balancing approach that incorporates an additional factor for more balanced and meaningful scoring.

Balancing Length Factor

To achieve a fair ranking, we needed a mechanism that dynamically adjusts the influence of TF-IDF scores and sentence length. This approach creates a flexible ranking mechanism, where the relative importance of each factor can be controlled to ensure an optimal trade-off between uniqueness and context.

To balance this bias and achieve a trade-off between the two extremes, the following formula was introduced:

$$\text{Score} = (\lambda_1 \cdot \text{Normalized\_TF\_IDF}) + (\lambda_2 \cdot \text{length})$$

$$\text{Where, } \lambda_1 = 1 - \lambda_2 \quad \text{and} \quad 0 \leq \lambda_1, \lambda_2 \leq 1$$

These are weights to control the relative importance of **Normalized_TF_IDF** and **length** in the final ranking.

- $\lambda_1 > \lambda_2$: Focus on sentences with unique terms (higher TF-IDF score).
- $\lambda_2 > \lambda_1$: Prioritize sentences with more context (lengthier ones).

This formula effectively balances the biases introduced by normalization and sentence length by distributing the total weight between the two factors. Since λ_1 = 1 - λ_2 , increasing the weight on one factor automatically reduces the influence of the other, ensuring a controlled trade-off. If λ_1 is higher, the ranking favors sentences with higher TF-IDF scores, emphasizing term uniqueness. Conversely, if λ_2 is higher, longer sentences with more contextual richness are prioritized. This dynamic weighting mechanism allows for fine-tuning based on the specific needs of the classification task, preventing extreme biases toward either short or long sentences.

4 Results and Discussion

4.1 Number of Sentences Approach

The results in Table 1 and Figure 4 show that accuracy improves as more sentences are selected, with the order: ranked > first > random > last. In the ranked method, accuracy peaks at three sentences. Selecting a fixed number of top-ranked sentences keeps inference time nearly constant and achieves **94.19%** accuracy with just 5 sentences only **0.544%** below the full-context baseline of 94.706%. This shows that substantial context reduction is possible with minimal accuracy loss. To further refine this, we introduce a normalization strategy that combines normalized TF-IDF scores with sentence length to capture both relevance and informational content.

Sentence(s)	First	Last	Random	Ranked
1	**90.70%**	81.64%	75.76%	90.35%
2	93.01%	87.60%	90.31%	**93.17%**
3	93.17%	89.76%	91.49%	**93.64%**
4	92.82%	91.09%	91.72%	**94.00%**
5	93.56%	91.64%	92.70%	**94.19%**

Table 1: Sentence-wise Accuracy Results — The table shows accuracy across different sentence selection strategies (first, last, random, ranked) for 1 to 5 selected sentences. Results indicate that the ranked approach performs best, followed by first, random, and last, highlighting the importance of the selection method and sentence count on model performance.

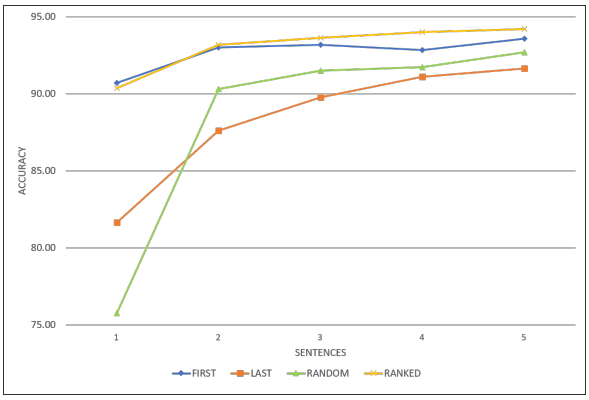

Figure 4: Sentence-wise accuracy graph — The graph visualizes sentence-wise accuracy for different selection methods: first, last, random, and ranked. It plots accuracy against the number of selected sentences (1 to 5). The graph shows that ranked > first > random > last.

Normalization Results

Table 2 shows that combining normalized TF-IDF scores with sentence length yields peak accuracy at λ_2 = 0.7, achieving **94.07%** with 4 sentences. Longer sentences prove more informative in minimal contexts, while λ_2 values of 0.2 and 0.7 best balance relevance and length. Accuracy stabilizes as sentence count increases.

Sentence(s)	0.2 (λ_2)	0.5 (λ_2)	0.7 (λ_2)	1.0 (λ_2)
1	89.11%	88.94%	89.10%	**89.26%**
2	**92.82%**	91.86%	92.73%	92.23%
3	93.79%	93.81%	93.78%	**93.82%**
4	93.95%	93.36%	**94.07%**	93.59%
5	93.47%	93.56%	**93.67%**	93.32%

Table 2: Normalized sentence-wise accuracy results — The table presents normalized sentence-wise accuracy results for the ranked sentence selection method. It shows results across different values of λ_2 ranging from 0.2 to 1.0. $\lambda_2 = 0.7$ provides optimal performance for different sentence counts.

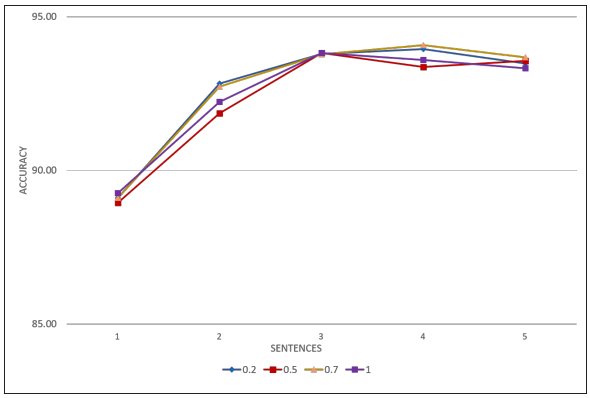

Figure 5: Normalized sentence-wise accuracy graph — The graph shows normalized sentence-wise accuracy for the ranked selection method. The x-axis represents the number of selected sentences, while the y-axis shows the corresponding accuracy. Each line corresponds to a different λ_2.

Sentence(s)	Ranked	Ranked Normalized
1	90.35%	89.11%
2	93.17%	92.82%
3	93.64%	93.82%
4	94.00%	94.07%
5	94.19%	93.67%

Table 3: Comparison of Ranked and Ranked-Normalized Results — The table compares the accuracy of sentence selection using ranked and ranked-normalized methods. It shows that normalization has little impact when the number of selected sentences is low.

Table 3 compares basic ranked selection with normalized ranking, showing minimal improvement when few sentences are selected.

4.2 Data Percentage Approach

Table 4 and Figure 6 illustrate the accuracy achieved by selecting first, last, random, and ranked percentages of sentences from documents. Using the full-length documents for and testing yields an

Percentage	First	Last	Random	Ranked	Ranked Normalized
10%	90.74%	71.76%	86.00%	**91.80%**	91.14%
20%	93.25%	87.88%	90.31%	93.41%	**93.64%**
30%	93.19%	90.31%	92.22%	93.29%	**93.80%**
40%	93.58%	91.96%	92.82%	93.98%	**94.39%**
50%	94.19%	92.98%	93.33%	**94.51%**	94.04%
60%	**94.31%**	92.86%	93.05%	**94.31%**	93.60%
70%	94.62%	94.19%	94.31%	93.92%	**94.47%**
80%	**94.43%**	93.45%	94.23%	94.15%	94.15%
90%	94.50%	93.56%	94.03%	94.11%	**94.90%**
100%	94.35%	94.35%	94.11%	94.63%	**94.78%**

Table 4: Percentage-wise accuracy results — The table presents percentage-wise accuracy results for sentence selection approaches at coverage levels from 10% to 100%. It compares First, Last, Random, Ranked, and Ranked Normalized methods. At 100% coverage, minor accuracy variations (<1%) arise from sentence reordering in Random and Ranked methods, whereas First and Last preserve the original order, yielding identical results.

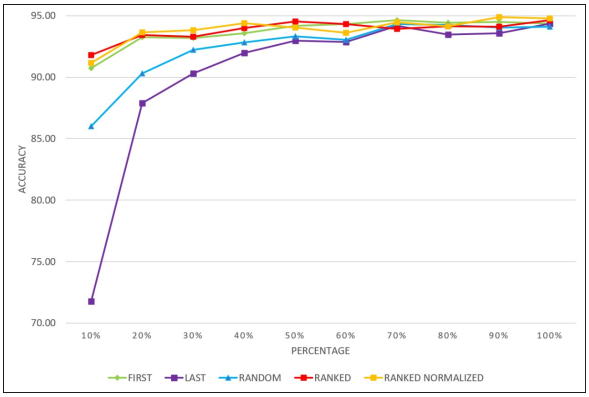

Figure 6: Percentage-wise accuracy graph — The graph visualizes percentage-wise accuracy for different sentence selection methods as sentence coverage increases from 10% to 100%. The x-axis represents the percentage of selected sentences, while the y-axis shows accuracy. Each line corresponds to a method: First, Last, Random, Ranked, and Ranked Normalized. The graph highlights how accuracy improves with more context and which methods are most effective.

accuracy of 94.706%, which serves as the baseline for comparison. At 100% coverage, accuracies are nearly identical across methods, with variations of less than 1%. These slight differences arise from sentence reordering in Random and Ranked selections, which alters the semantic flow and impacts model interpretation, whereas First and Last preserve the original order, yielding identical results. Importantly, by reducing the context to just 40 to

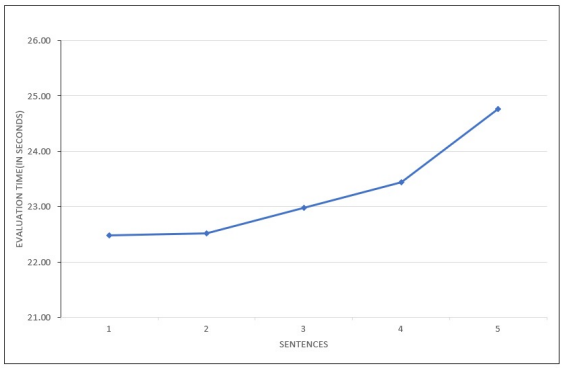

Figure 7: Evaluation time graph for sentence-wise selection — The graph represents the relationship between evaluation time (in seconds) and the number of sentences considered during sentence-wise selection. The x-axis denotes the number of sentences, while the y-axis shows the corresponding evaluation time required. The graph typically highlights a trend where evaluation time increases as the number of sentences grows

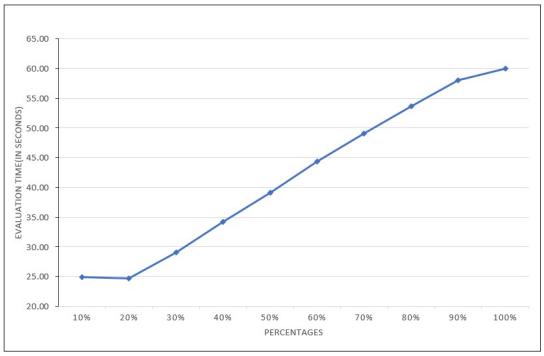

Figure 8: Evaluation time graph for percentage-wise selection — The graph illustrates the relationship between evaluation time (in seconds) and the percentage of sentences selected during percentage-wise selection. The graph demonstrates a trend where evaluation time changes based on the percentage selected.

50 percent of the original document, we are still able to achieve an impressive accuracy of **94.39%**, remarkably close to the base accuracy of 94.706%, demonstrating that its performance is competitive with approaches that use the full document context. With reduced context sizes, the Ranked Selection method consistently outperforms other techniques, such as First, Last, and Random selection. As the context size increases, the performance of all methods converges, yielding similar results. This convergence indicates that the ranked selection method is particularly effective in enhancing accuracy when operating with smaller context windows. In this setting, normalization shows a positive impact, enhancing performance in most cases.

4.3 Inference Time

Context reduction aims to minimize inference time while preserving accuracy. Using TF-IDF, we dynamically adjust sequence length to match document content, avoiding inefficiencies from fixed limits like BERT's 512 tokens. Figures 7 and 8 show testing time variations across different context lengths.

Figure 7 shows a positive correlation between the number of sentences and evaluation time. While the increase is modest from 1 to 3 sentences, it becomes more pronounced from sentence 4 onward, indicating that evaluation time grows increasingly with higher sentence counts.

Figure 8 shows evaluation time stays stable from 10–20%, then rises sharply from 30%, peaking at 100%. At 40% context, our method reaches 94.39% accuracy, only 0.33% below the full-context baseline, while cutting inference latency by 43%. This highlights an efficient speed–accuracy trade-off for scalable, real-world use.

5 Conclusion

We propose an efficient approach to long document classification using sentence selection techniques that reduce input size while maintaining accuracy comparable to full-context models. Strategies include first/last sentence selection, random sampling, and TF-IDF-based ranking.

On the Marathi LDC dataset from L3Cube's IndicNews collection, our method significantly reduces computational costs without sacrificing performance, with TF-IDF ranking proving especially effective. We examined the trade-off between input size and accuracy, finding that selecting a proportion of high-ranking sentences yields better efficiency–performance balance than a fixed number, and that normalization can further improve results. Overall, our approach is scalable, resource-efficient, and adaptable, with potential for domain-specific selection or hybrid models to refine input representation in future work.

6 Limitations

While our MahaBERT-based model captures deep semantics from selected sentences, the TF-IDF based selection relies solely on term frequency,

ignoring contextual relationships.
Future work could incorporate semantic embeddings to improve relevance and reduce redundancy. Additionally, discourse parsers or coherence models may help maintain logical flow, preserve essential context, and enhance interpretability.

Acknowledgement

This work was undertaken with the mentorship of L3Cube, Pune. We sincerely appreciate the invaluable guidance and consistent encouragement provided by our mentor during this endeavor.

References

Mirashi Aishwarya, Sonavane Srushti, Lingayat Purva, Padhiyar Tejas, and Joshi Raviraj. 2023. L3cube-indicnews: News-based short text and long document classification datasets in indic languages. In *Proceedings of the 20th International Conference on Natural Language Processing (ICON)*, pages 442–449.

Muhammad Al-Qurishi. 2022. Recent advances in long documents classification using deep-learning. In *Proceedings of the 5th International Conference on Natural Language and Speech Processing (ICNLSP 2022)*.

David Bamman and Noah Smith. 2013. New alignment methods for discriminative book summarization.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer.

Bijoyan Das and Sarit Chakraborty. 2020. An improved text sentiment classification model using tf-idf and next word negation.

Mamata Das, K. Selvakumar, and P.J.A. Alphonse. 2023. A comparative study on tf-idf feature weighting method and its analysis using unstructured dataset.

J. Devlin, M. W. Chang, K. Lee, and K. Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Jun He. 2019. Long document classification from local word glimpses via recurrent attention learning. *IEEE Access*.

Kushal Jain, Adwait Deshpande, Kumar Shridhar, and 1 others. 2020. Indic-transformers: An analysis of transformer language models for indian languages.

Raviraj Joshi. 2022. L3cube-mahacorpus and mahabert: Marathi monolingual corpus, marathi bert language models, and resources. In *Proceedings of the WILDRE-6 Workshop within the 13th Language Resources and Evaluation Conference*, pages 97–101.

Snehal Ishwar Khandve, Vedangi Kishor Wagh, Apurva Dinesh Wani, Isha Mandar Joshi, and Raviraj Bhuminand Joshi. 2022. Hierarchical neural network approaches for long document classification. In *Proceedings of the 2022 14th International Conference on Machine Learning and Computing*, pages 115–119.

Sang-Woon Kim and Joon-Min Gil. 2019. Research paper classification systems based on tf-idf and lda schemes. *Human-centric Computing and Information Sciences*.

Chao Li, Yanfen Cheng, and Hongxia Wang. 2018. A novel document classification algorithm based on statistical features and attention mechanism.

C. z. Liu, Y. x. Sheng, Z. q. Wei, and Y. Q. Yang. 2018a. Research of text classification based on improved tfi-df algorithm. In *2018 IEEE International Conference of Intelligent Robotic and Control Engineering (IRCE)*.

L. Liu, K. Liu, Z. Cong, J. Zhao, Y. Ji, and J. He. 2018b. Long length document classification by local convolutional feature aggregation. *Algorithms*.

André F. T. Martins, António Farinhas, Marcos Treviso, and 1 others. 2020. Sparse and continuous attention mechanisms.

S. Minaee, N. Kalchbrenner, E. Cambria, and 1 others. 2021. Deep learning based text classification: A comprehensive review.

Saloni Mittal, Vidula Magdum, Sharayu Hiwarkhedkar, Omkar Dhekane, and Raviraj Joshi. 2023. L3cube-mahanews: News-based short text and long document classification datasets in marathi. In *International Conference on Speech and Language Technologies for Low-resource Languages*, pages 52–63. Springer.

G. Moro. 2023. Efficient memory-enhanced transformer for long-document summarization in low-resource regimes. *Sensors*.

Hyunji Park, Yogarshi Vyas, and Kashif Shah. 2022. Efficient classification of long documents using transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*.

Linh Manh Pham and Hoang Cao The. 2024. Lnlf-bert: Transformer for long document classification with multiple attention levels. *IEEE Access*.

Sumanth Prabhu, Moosa Mohamed, and Hemant Misra. 2021. Multi-class text classification using bert-based active learning.

Bella Adriani Putri and Erwin Budi Setiawan. 2023. Topic classification using the long short-term memory (lstm) method with fasttext feature expansion on twitter.

Shahzad Qaiser and Ramsha Ali. 2018. Text mining: Use of tf-idf to examine the relevance of words to documents. *IJCA*.

B. Song. 2024. State space models based efficient long documents classification. *Journal of Intelligent Learning Systems and Applications*.

Chi Sun, Xipeng Qiu, Yige Xu, and 1 others. 2020. How to fine-tune bert for text classification?

Y. Tay, M. Dehghani, S. Abnar, and 1 others. 2021. Long range arena: A benchmark for efficient transformers.

Shoryu Teragawa, Lei Wang, and Ruixin Ma. 2021. A deep neural network approach using convolutional network and long short term memory for text sentiment classification. In *2021 IEEE 24th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*.

Vedangi Wagh, Snehal Khandve, Isha Joshi, Apurva Wani, Geetanjali Kale, and Raviraj Joshi. 2021. Comparative study of long document classification. In *TENCON 2021-2021 IEEE Region 10 Conference (TENCON)*, pages 732–737. IEEE.

Zichao Yang, Diyi Yang, Chris Dyer, and 1 others. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

O. Zafrir. 2019. Q8bert: Quantized 8bit bert.

M. Zaheer, J. Ainslie, G. Guruganesh, and 1 others. 2020. Big bird: Transformers for longer sequences. In *Proceedings of NeurIPS*, pages 702–709.