Jurnal CoreIT, Vol.11, No.2, December 2025
ISSN 2460-738X (Print)
ISSN 2599-3321 (Online)

Combining BERT and Graph-Based Ranking for Extractive Summarization of Indonesian News Articles

I Nyoman Prayana Trisna[1*], Wayan Oger Vihikan[2], Anis Zahra Nur Azizah[3]
[123]Udayana University
prayana.trisna@unud.ac.id*, oger_vihikan@unud.ac.id, zahranurazizah@student.unud.ac.id

Abstract. Automatic text summarization is an effective solution to manage the vast amount of information in the digital age. This study aims to develop an extractive text summarization system for Indonesian news articles using sentence embeddings generated by IndoBERT and mBERT, combined with TextRank and LexRank algorithms for sentence ranking. The dataset used is Indonesian Text Summarization (IndoSum), which contains thousands of manually summarized articles. The research includes data collection, cleaning, preprocessing, embedding extraction, sentence similarity calculation, and ranking using graph-based methods. Model performance was evaluated using ROUGE and BERTScore. The results show that the combination of IndoBERT and LexRank achieved the highest performance with ROUGE-1 score 0.7018 and BERTscore 0.8696. Compared to the baseline model from the origin of the dataset, our approach surpasses in the ROUGE metrics, showing the capability of combining embedding approach and graph-based ranking for effective extractive text summarization for Indonesian news article.

Keywords: Extractive Summarization, News Article, BERT, LexRank, TextRank

Received September 2025 / **Revised** December 2025 / **Accepted** December 2025

INTRODUCTION

The rapid development of the internet has driven the growth of online media sites in Indonesia [1]. Reading is an inseparable part of human life. However, most people are not interested in reading long texts and tend to skip crucial parts [2]. Automatic summarization can be a solution to this problem. Automatic summarization produces a summary containing important sentences and includes all relevant information from the original document [3]. Automatic summarization aims to transform text input into a concise form to present the most important information to users [4]. Summarization techniques can be classified into two main categories: extractive and abstractive. Extractive summarization has become a significant area of research, particularly in the context of news articles.

One approach to text summarization is a graph-based algorithm that uses a graph structure to model the relationships between elements in a data set and performs a ranking or decision-making process based on that structure. PageRank is a graph-based ranking algorithm used to rank the importance of a web page in a hyperlink network on the Internet [5]. PageRank is generally applied to directed graphs, but in the case of undirected graphs, the PageRank algorithm can also be used with modifications similar to those used in TextRank [6]. The TextRank algorithm is a graph-based ranking algorithm for scoring text, where text units can be defined as keywords or sentences [7]. This algorithm was introduced as a completely unsupervised approach. Besides TextRank, LexRank is a graph-based algorithm that can be used for extractive summarization. LexRank assesses the importance of a sentence based on its centrality within a sentence network [8]. Both TextRank and LexRank assume the document as a form of sequences of vector.

Graph-based methods have been successfully applied to real-world datasets, such as in research focused on Indonesian text summarization. Kurniawan and Louvan [9] introduced a publicly accessible dataset for summarizing Indonesian text called IndoSum. The research was conducted using various approaches, including unsupervised approaches. In the unsupervised approach, the LexRank method demonstrated superior performance, with a ROUGE-1 score reaching 0.35. Another study conducted on a similar dataset using a combination of the LexRank and YAKE algorithms resulted in an increase in the ROUGE-1 score to 0.453 [10]. Both studies focused on basic techniques such as unsupervised method, whereas this study does not only use unsupervised method, but also incorporate transformers on the IndoSum dataset.

DOI: 10.24014/coreit.v11i2.37929

Transformers are a type of neural network that have revolutionized various fields [11], with recent research increasingly turning toward advanced pre-trained model like the Bidirectional Encoder Representations from Transformers (BERT) model [12]. The BERT model can be used for extractive text summarization. BERT's strength lies in generating high-quality sentence embeddings, which effectively capture the content and context of each sentence, a crucial step for subsequent ranking for extractive text summarization. Beside the base version, BERT has another variant designed to handle multiple languages simultaneously called Multilingual BERT (mBERT) [13]. Centralized training for multiple languages in mBERT aims to facilitate cross-language transfer for NLP tasks. On the other hand, BERT can be orchestrated for specific-language. For instance, IndoBERT is a pre-trained model using the BERT algorithm for Bahasa Indonesia [14].

In recent years, the BERT model has become increasingly dominant in addressing complex NLP tasks [15]. Its broad capability is demonstrated in various applications, including classifying emerging industries and demonstrates impressive precision in categorizing business descriptions, achieving accuracy rates ranging from 84.11% to 99.66% across 16 industry classifications [16]. Another study shows the capability of using BERT to analyze sentiment on the impact of the coronavirus on social life, yielded a high validation accuracy of 94% [17]. A study comparing the performance of the IndoBERT and mBERT models was also conducted by [18] in detecting Indonesian-language hoaxes, depicting the capability of multilingual and language-specific model. Bano et al. [19] proposed an innovative approach for extractive summarization using BERT and BiGRU on the arXiv and PubMed datasets containing long scientific documents. Research on extractive summarization in languages other than English was also conducted by Nada et al. [20], who used the AraBERT model to summarize Arabic text and demonstrated good performance with a ROUGE-2 score of 0.51. Research in the Indonesian language has also been conducted by [21] on court decisions related to narcotics cases, utilizing the IndoBERT model, which has yielded significant ROUGE scores. Research on Indonesian language articles has also been conducted by [22] on the Liputan6 dataset using IndoBERT embedding in the TextRank algorithm, resulting in a ROUGE-1 score of 0.3929. Based on various previous studies, the BERT model has been shown to be capable of producing effective text summaries, even for long documents and in various non-English languages. These various previous studies confirm the capability of the BERT model to produce effective text summaries, even for long documents and across various non-English languages, underscoring its suitability for integration into a robust extractive summarization pipeline.

In line with the transformer revolution in the field of NLP, this study intends to conduct research combining the use of transformers such as IndoBERT and mBERT with unsupervised algorithms such as LexRank and TextRank. The integration between transformer-based embeddings and graph-based ranking is hypothesized to improve overall metrics compared to isolated approaches. The BERT model was chosen because the mBERT model was trained on multiple languages, and the IndoBERT model was trained on Indonesian, which is suitable for the dataset used in this study. The goal of this study is to build an extractive summarization model for Indonesian language articles based on BERT and PageRank. This research is expected to improve the quality of extractive-based automatic summarization for Indonesian language articles, making it easier for readers to access information.

METHODS

The research method includes literature studies, problem formulation, research objectives, data collection, model creation, and evaluation, as well as conclusions and suggestions. This research specifically does not explicitly split the dataset into training and testing data, since the following methods (embedding and graph-ranking) act as unsupervised methods. Instead, this study utilizes both training and testing data for the evaluation, with the result on testing data will be compared to previous study with similar dataset. The modeling design flow can be seen in Figure 1 below.

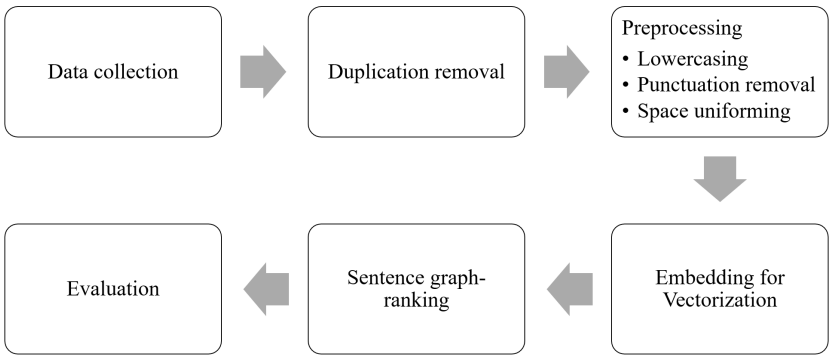

Figure 1. Research flow

The data collection stage is the initial stage of the research. The data used in this study comes from The Indonesian Text Summarization (IndoSum), which is publicly accessible on Kaggle[1]. The IndoSum dataset contains 18,774 articles with each corresponding provided summary, provided by Shortir, an Indonesian news aggregator and summary company [9]. The dataset is already split into 3 parts: training, validation, and test.

The data cleaning phase is essential for the next step. This process includes checking for duplication. If duplicate data are found, they are removed to prevent them from affecting the results. Then, the preprocessing stage was carried out to eliminate inconsistencies in the data. The preprocessing performed in this study included lowercasing the text, removing punctuation, and applying uniform spacing. The preprocessing, specifically the lowercasing part, aims to reduce the number of vocabularies for better generalization in the embedding process. Additionally, we limit the number of maximum sentences in the data by 4, thus the summary data with longer sentences are removed.

The embedding process is conducted after making sure the data is ready for modelling. The embedding models used are IndoBERT and mBERT as stated before. Data that has gone through the preprocessing stage is vectorized using the built-in tokenizers in IndoBERT and mBERT without further fine-tuning for both models. This process aims to represent each sentence in vector form. The output of this process is used to calculate the similarity between sentences. These tokenizers are used to convert raw text into numerical representations that the BERT model can process. Data that has passed this stage will then proceed to the next stage, namely sentence ranking and summary generation.

After the embedding process, the sentence ranking takes place. Graph-based ranking approaches using LexRank and TextRank are used to calculate the importance of sentences contained in the text. Both algorithms work by utilizing a similarity matrix using the cosine similarity of the sentence embedding vectors generated in the previous process.

In the TextRank algorithm, the system constructs an undirected weighted graph from the similarity matrix and then applies the PageRank algorithm to assign an importance score to each sentence based on the connections and weights between nodes. In brief, the PageRank is explained in Equation 1.

$$PR(u) = \frac{1-d}{N} + d \sum_{v \in B_u} \frac{PR(v)}{L(v)} \quad (1)$$

Where $PR(u)$ is the PageRank score for the sentence u, N is the total number of nodes in the graph, B_u is the set of nodes linking to u, and $L(v)$ is the sum of outgoing weight from node v. PageRank can be calculated iteratively using the power method until convergence is achieved, when the difference in

[1] https://www.kaggle.com/datasets/linkgish/indosum

PageRank values between iterations is very small. The sentences with the highest scores are then selected as part of the extractive summary.

Meanwhile, the LexRank algorithm involves forming a matrix from the similarity matrix with or without a threshold. LexRank is notated in Equation 2.

$$PR(u) = \frac{1-d}{N} + d \sum_{v \in Adj(u)} \left(\frac{PR(v)}{|Adj(v)|} \right) \quad (2)$$

Similar to Equation 1, the LexRank is conceptually similar to the TexRank. The difference is lies in the outgoing weight. In TextRank, $L(v)$ considers all the weight of outgoing links, while in LexRank, $Adj(v)$ only considers adjacent links that bypass a certain threshold. This study uses a threshold value of 0.2 to eliminate relationships between sentences that are deemed insufficiently similar. The sentences with the highest scores are then selected as part of the extractive summary.

This study will evaluate various combinations of IndoBERT or mBERT embeddings with TextRank or LexRank ranking algorithms, measuring performance in summary generation using ROUGE and BERTScore. ROUGE works by measuring the degree of n-gram overlap between model-generated text and original human-written text [22]. A higher score indicates a greater degree of similarity between the two texts. In this study, the ROUGE-N and ROUGE-L metrics were used to evaluate system performance. ROUGE-N measures the number of n-gram matches between the system-generated summary *S* and the provided summary from the dataset *R*, while ROUGE-L evaluates the summary based on the longest common subsequence (LCS) between generated summary *S* and the provided summary from the dataset *R*. Respectively, as written in Equation 3 and 4.

$$\text{ROUGE-N} = \frac{\sum_{\text{R}} \sum_{n\text{-gram} \in \text{S}} \text{Count}_{\text{match}}(n\text{-gram})}{\sum_{\text{R}} \sum_{n\text{-gram} \in \text{S}} \text{Count}(n\text{-gram})} \quad (3)$$

$$\text{ROUGE-L} = \frac{2 \cdot \text{LCS}(S, R)}{\text{length}(S) + \text{length}(R)} \quad (4)$$

Meanwhile, BERTscore [23] is an evaluation metric that utilizes the BERT model to calculate the similarity score between tokens. The BERTScore evaluates a summary by generating contextualized token embeddings from BERT for both the system-generated summary and the provided summary from the dataset. Precision is then calculated by measuring how much of the generated summary's content is semantically present in the provided summary, as demonstrated in Equation 5. *T* is the tokens in the generated summary *x*, while *T'* is the token in the provided summary, and *e* is the set of contextualized token embeddings. Conversely, recall –as written in Equation 6, determines how much of the provided summary's content is captured by the generated summary. Then, the F1 score (depicted in Equation 7) is derived as the harmonic mean of precision and recall, serving as overall measure of the semantic quality.

$$\text{BERT}_{\text{P}} = \frac{1}{|T|} \sum_{x_i \in x} \max_{y_j \in y} \left(e_i^{\top} e_j \right) \quad (5)$$

$$\text{BERT}_{\text{R}} = \frac{1}{|T'|} \sum_{y_j \in y} \max_{x_i \in x} \left(e_i^{\top} e_j \right) \quad (6)$$

$$\text{BERT}_{\text{F1}} = 2 \cdot \frac{\text{BERT}_{\text{P}} \cdot \text{BERT}_{\text{R}}}{\text{BERT}_{\text{P}} + \text{BERT}_{\text{R}}} \quad (7)$$

The F1 score is utilized as the primary metric of BERTscore, as it represents the harmonic mean of precision and recall, thereby providing a single measure that symmetrically weights the importance of both.

This research integrates graph-based ranking methodologies with BERT-based sentence embeddings. For the implementation, *indobert-base-p2* is utilized for IndoBERT, while *bert-base-multilingual-cased* represents the mBERT variant. Both pre-trained models feature a consistent architecture of 12 hidden

layers, a hidden size of 768 dimensions, and 12 attention heads. IndoBERT is trained exclusively on Indonesian text with a vocabulary of 31,923 unique tokens [24], whereas mBERT employs a shared vocabulary across 104 languages comprising 119,547 unique tokens [25]. These models are evaluated in combination with the TextRank and LexRank algorithms; for the LexRank configuration, a similarity threshold of 0.2 is established to determine vertex adjacency. Additionally, both training and testing data will be employed for evaluation.

RESULT AND DISCUSSION

The result for all experiments on training data is depicted in Table 1, and on testing data are presented in Table 2 alongside the benchmark findings previously established by Kurniawan and Louvan [9]. The study by Kurniawan and Louvan did not employ BERTscore as evaluation metric.

ROUGE scores are computed individually for each generated summary relative to its corresponding provided summary, and the mean is subsequently calculated. Similarly, F1 BERTScore is calculated to evaluate each generated summary semantically based on Equation 7, then the score results are subsequently averaged.

Table 1. Evaluation scores on training data

Scenario	ROUGE-1	ROUGE-2	ROUGE-L	$BERT_{F1}$
IndoBERT + TextRank	0.3967	0.2656	0.2887	0.7475
IndoBERT + LexRank	**0.7018**	**0.6645**	**0.6832**	**0.8696**
mBERT + TextRank	0.4133	0.2887	0.3054	0.7534
mBERT + LexRank	0.6986	0.6602	0.6783	0.8680

The results on training data and testing data are presented in Table 1 and Table 2, respectively. Notably, the difference in result scores in both tables is negligible, remarking the robustness of the proposed models. Additionally, this indicates the ability of models to generalize and learn the patterns for text summarization.

Based on Table 1 and Table 2, the performance of IndoBERT embedding in the LexRank algorithm shows better results compared to other models. This indicates that the sentence representation generated by IndoBERT effectively captures the semantic context of sentences. Furthermore, the graph-based LexRank algorithm can evaluate the global importance of sentences by calculating the similarity between weighted sentences. This combination produces superior performance compared to other methods.

Table 2. Evaluation scores on testing data, compared to previous study

Scenario	ROUGE-1	ROUGE-2	ROUGE-L	BERTscore
IndoBERT + TextRank	0.4005	0.2686	0.2909	0.7475
IndoBERT + LexRank	**0.6990**	**0.6608**	**0.6795**	**0.8696**
mBERT + TextRank	0.4151	0.2902	0.3070	0.7534
mBERT + LexRank	0.6958	0.6563	0.6746	0.8680
NeuralSum [9]	0.6760	0.6116	0.6686	-
NeuralSum (300 emb. size) [9]	0.6790	0.6165	0.6724	-
NeuralSum + FastText [9]	0.6778	0.6137	0.6705	-

Based on the ROUGE score at Table 1 and Table 2, the LexRank method outperforms the TextRank with significant differences. On the contrary, the distinction in ROUGE score between IndoBERT and mBERT is slight. This indicates that the graph-based ranking method is more prominent than the transformed-based vectorization method for automatic extractive summarization. Thus, for extractive-based summarization, ROUGE score is quite influential to determine the model's overall performance since it measures the model's ability to extract the main sentence sequence by sequence.

However, the F1 BERTscore for each model cannot be overlooked, since it reveals how well the model transcribes overall context of the whole text into summary based on the similarity score between extracted tokens and provided tokens. In both Table 1 and 2, the performance difference between each experimented

model is quite apparent, especially when the ranking methods are taken into consideration. Although the metric difference is not as significant as with ROUGE score, it is evident that LexRank outperforms TextRank in term of text summarization.

The LexRank algorithm demonstrates superior performance in generating high-quality summaries compared to TextRank, regardless of the type of embedding used. TextRank relies on vector representations formed from word frequencies and only captures surface relationships between sentences. LexRank, on the other hand, calculates similarity between sentences against contextual embeddings and constructs a graph based on global relationships between sentences. Therefore, the use of transformer-based embeddings is more suitable for the LexRank algorithm than TextRank.

On the embedding side, the IndoBERT model demonstrated superior performance compared to mBERT, especially when combined with the LexRank algorithm. IndoBERT was specifically trained on an Indonesian language corpus, enabling it to more accurately capture local language structure, vocabulary, and style. Meanwhile, mBERT is generalist and not fully optimized for a specific language. Although, mBERT still produced quite good results, its ROUGE and BERTscore scores tended to be slightly lower than those of IndoBERT. This suggests that using embeddings specifically trained for the target language provides an advantage in understanding meaning and sentence structure in extractive contexts. This finding aligns with the result obtained by Tobing et.al [14] which states that specific-language BERT performs better than multilingual BERT.

Subsequently, we try to compare our best result with the experiment by Kurniawan and Louvan [9] in Table 1. Our best approach, IndoBERT with LexRank, outperforms the highest reported NeuralSum across all ROUGE metrics. This result collectively establishes a new state-of-the-art for the summarization task in Indonesian, highlighting the effectiveness of integrating contextual embedding with graph-based ranking method for extractive summarization.

CONCLUSION

This study develops an extractive summarization model utilizing BERT model embeddings, specifically IndoBERT and multilingual BERT (mBERT), in conjunction with graph-based TextRank and LexRank algorithms. The processes carried out include data collection, data cleaning, preprocessing, embeddings, sentence ranking and summary formation, evaluation, and system implementation. The data used in this study comes from Indonesian Text Summarization (IndoSum), which is publicly accessible on Kaggle. The results obtained from this study indicate that the application of IndoBERT embedding in the LexRank algorithm shows the best performance in extractive text summarization on Indonesian articles with ROUGE-1 reaching 0.7018, ROUGE-2 reaching 0.6645, ROUGE-L reaching 0.6832, and BERT score of 0.8696 on the given training data. This indicates that the use of embeddings explicitly trained for the target language provides advantages in understanding the meaning and structure of sentences in extractive contexts. The LexRank algorithm demonstrated superior performance in producing high-quality summaries compared to TextRank, regardless of the type of embedding used. Therefore, the use of transformer-based embedding is more suitable for the LexRank algorithm than TextRank. In addition, our combined approaches outperform the base model in the dataset. For further development, it is recommended to explore different sentence counts in the summary to determine their impact on the quality of the resulting summary.

REFERENCES

[1] M. R. Hadwirianto, F. Hamami, and O. N. Pratiwi, "Extractive Text Summarization Terhadap Artikel Berita Indonesia Berbasis Machine Learning," eProceedings of Engineering, vol. 11, no. 4, 2024.

[2] R. Bhargava and Y. Sharma, "Deep extractive text summarization," Procedia Computer Science, vol. 167, pp. 138–146, 2020.

[3] A. P. Widyassari, S. Rustad, G. F. Shidik, E. Noersasongko, A. Syukur, A. Affandy, and D. R. I. M. Setiadi, "Review of automatic text summarization techniques & methods," Journal of King Saud University-Computer and Information Sciences, vol. 34, no. 4, pp. 1029–1046, 2022.

[4] M. Asmitha, A. Danda, H. Bysani, R. P. Singh, and S. Kanchan, "Automation of text summarization using Hugging Face NLP," in *2024 5th International Conference for Emerging Technology (INCET)*, May 2024.

[5] T. Page, Lawrence; Brin, Sergey; Motwani, Rajeev; Winograd, "The PageRank Citation Ranking: Bringing Order to the Web," Stanford InfoLab, 442, Nov. 1999. [Online]. Available: http://ilpubs.stanford.edu:8090/422/1/1999-66.pdf

[6] A. Kazemi, V. Pérez-Rosas, and R. Mihalcea, “Biased TextRank: Unsupervised graph-based content extraction,” in Proc. of the 2020 Conference on Computational Linguistics (COLING), Dec. 2020, pp. 1599–1611.
[7] R. Mihalcea and P. Tarau, “TextRank: Bringing order into texts,” Proc. 2004 Conf. Empir. Methods Nat. Lang. Process. EMNLP 2004 - A Meet. SIGDAT, a Spec. Interes. Gr. ACL held conjunction with ACL 2004, vol. 85, pp. 404–411, 2004.
[8] D. R. Radev, “LexRank: Graph-based Lexical Centrality as Salience in Text Summarization,” J. Artif. Intell. Res., vol. 22, pp. 457–479, 2004.
[9] K. Kurniawan and S. Louvan, “IndoSum: A New Benchmark Dataset for Indonesian Text Summarization,” Proc. 2018 Int. Conf. Asian Lang. Process. IALP 2018, pp. 215–220, 2018.
[10] J. Wijaya and A. S. Girsang, “Indonesian News Extractive Summarization using Lexrank and YAKE Algorithm,” Stat. Optim. Inf. Comput., vol. 12, no. 6, pp. 1973–1983, 2024.
[11] I. Vaswani, Ashish; Shazeer, Noam; Parmar, Niki; Uszkoreit, Jakob; Jones, Llion; Gomez, Aidan N.; Kaiser, Lukasz; Polosukhin, “Attention is All You Need,” in Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, New York, NY, USA: ACM, Oct. 2023, pp. 4752–4758.
[12] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” NAACL HLT 2019 - 2019 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. - Proc. Conf., vol. 1, no. Mlm, pp. 4171–4186, 2019.
[13] T. Pires, E. Schlinger, and D. Garrette, “How multilingual is multilingual BERT?,” in Proc. 57th Annu. Meeting Assoc. Comput. Linguistics, 2019, pp. 4996–5001.
[14] U. Khairani, V. Mutiawani, and H. Ahmadian, “Pengaruh Tahapan Preprocessing Terhadap Model Indobert Dan Indobertweet Untuk Mendeteksi Emosi Pada Komentar Akun Berita Instagram,” J. Teknol. Inf. dan Ilmu Komput., vol. 11, no. 4, pp. 887–894, 2024.
[15] A. D. Widiantoro and S. Ridwan, PENGANTAR NLP DAN TOPIK MODEL LDA. Palembang, Indonesia: Asosiasi Doktor Sistem Informasi Indonesia, 2024.
[16] B. Yang, B. Zhang, K. Cutsforth, S. Yu, and X. Yu, “Emerging industry classification based on BERT model,” Inf. Syst., vol. 128, no. October 2024, 2025.
[17] M. Singh, A. K. Jakhar, and S. Pandey, “Sentiment analysis on the impact of coronavirus in social life using the BERT model,” Soc. Netw. Anal. Min., vol. 11, no. 1, pp. 1–11, 2021.
[18] C. J. L. Tobing, IGN Lanang Wijayakusuma, and Luh Putu Ida Harini, “Perbandingan Kinerja IndoBERT dan MBERT Untuk Deteksi Berita Hoaks Politik dalam Bahasa Indonesia,” JST (Jurnal Sains dan Teknol., vol. 14, no. 1, pp. 114–123, 2025.
[19] S. Bano, S. Khalid, N. M. Tairan, H. Shah, and H. A. Khattak, “Summarization of scholarly articles using BERT and BiGRU: Deep learning-based extractive approach,” J. King Saud Univ. - Comput. Inf. Sci., vol. 35, no. 9, p. 101739, 2023.
[20] A. M. Abu Nada, E. Alajrami, A. A. Al-Saqqa, and S. S. Abu-Naser, “Arabic Text Summarization Using AraBERT Model Using Extractive Text Summarization Approach,” Int. J. Acad. Inf. Syst. Res., vol. 4, no. 8, pp. 6–8, 2020, [Online]. Available: www.ijeais.org/ijaisr
[21] G. W. Wicaksono, S. F. Al asqalani, Y. Azhar, N. P. Hidayah, and A. Andreawana, “Automatic Summarization of Court Decision Documents Over Narcotic Cases Using BERT,” JOIV: International Journal on Informatics Visualization, vol. 7, no. 2, pp. 416–422, 2023.
[22] E. Yulianti, N. Pangestu, and M. A. Jiwanggi, “Enhanced TextRank using weighted word embedding for text summarization,” Int. J. Electr. Comput. Eng., vol. 13, no. 5, pp. 5472–5482, 2023.
[23] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, “BERTScore: Evaluating Text Generation with BERT,” in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Virtual Event, 2020
[24] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained Language Model for Indonesian NLP," in *Proc. 28th Int. Conf. Comput. Linguistics*, Barcelona, Spain, 2020, pp. 757–770.
[25] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proc. 2019 Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Language Technol.*, vol. 1, Minneapolis, MN, 2019, pp. 4171–4186.