Accredited SINTA 2 Ranking
Decree of the Director General of Higher Education, Research, and Technology, No. 158/E/KPT/2021
Validity period from Volume 5 Number 2 of 2021 to Volume 10 Number 1 of 2026

Published online at: **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 9 No. 2 (2025) 250 - 257 e-ISSN: 2580-0760

Comparing Word Representation BERT and RoBERTa in Keyphrase Extraction using TgGAT

Novi Yusliani[1*], Aini Nabilah[2], Muhammad Raihan Habibullah[3], Annisa Darmawahyuni[4], Ghita Athalina[5]
[1,2,3]Department of Informatic Engineering, Faculty of Computer Science, Universitas Sriwijaya, Palembang, Indonesia
[4,5]Department of Computer Engineering, Faculty of Computer Science, Universitas Sriwijaya, Palembang, Indonesia
[1]novi_yusliani@unsri.ac.id, [2]aininabilahalfatah@gmail.com, [3]muhraihanh23@gmail.com,
[4]riset.annisadarmawahyuni@gmail.com, [5]ghitaathalina@unsri.ac.id

Abstract

In this digital era, accessing vast amounts of information from websites and academic papers has become easier. However, efficiently locating relevant content remains challenging due to the overwhelming volume of data. Keyphrase Extraction Systems automate the process of generating phrases that accurately represent a document's main topics. These systems are crucial for supporting various natural language processing tasks, such as text summarization, information retrieval, and representation. The traditional method of manually selecting key phrases is still common but often proves inefficient and inconsistent in summarizing the main ideas of a document. This study introduces an approach that integrates pre-trained language models, BERT and RoBERTa, with Topic-Guided Graph Attention Networks (TgGAT) to enhance keyphrase extraction. TgGAT strengthens the extraction process by combining topic modelling with graph-based structures, providing a more structured and context-aware representation of a document's key topics. By leveraging the strengths of both graph-based and transformer-based models, this research proposes a framework that improves keyphrase extraction performance. This is the first to apply graph-based and PLM methods for keyphrase extraction in the Indonesian language. The results revealed that BERT outperformed RoBERTa, with precision, recall, and F1-scores of 0.058, 0.070, and 0.062, respectively, compared to RoBERTa's 0.026, 0.030, and 0.027. The result shows that BERT with TgGAT obtained more representative keyphrases than RoBERTa with TgGAT. These findings underline the benefits of integrating graph-based approaches with pre-trained models for capturing both semantic relationships and topic relevance.

Keywords: *Keyphrase Extraction, BERT, RoBERTa, Pre-Trained Language Models, Topic-Guided Graph Attention Networks*

How to Cite: Novi Yusliani, Aini Nabilah, Muhammad Raihan Habibullah, Annisa Darmawahyuni, and Ghita Athalina, "Comparing Word Representation BERT and RoBERTa in Keyphrase Extraction using TgGAT", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 9, no. 2, pp. 250 - 257, Mar. 2025.
DOI: https://doi.org/10.29207/resti.v9i2.6279

1. Introduction

In the digital era, the rapid growth of textual data presents both opportunities and challenges in information retrieval. While numerous platforms facilitate document access, the overwhelming volume of data makes it difficult to extract relevant content efficiently. Keyphrase extraction plays a crucial role in summarizing and indexing documents, yet existing methods still face limitations in accuracy and efficiency, particularly in handling implicit keyphrases and contextual nuances. Keyphrase extraction in documents is still done manually, which is inefficient and often does not accurately represent the content [1]. The use of keyphrase extraction is very important to support or improve the quality of downstream tasks, such as text representation processes, information retrieval, text summarization, and many others [2].

Keyphrase extraction plays a crucial role in summarizing the content and main topic of a document. Currently, there are two primary methods for keyword extraction: supervised and unsupervised, each with its unique challenges. Supervised methods often struggle with the lack of sufficient labeled data, as the labeling process is both time-consuming and costly. Meanwhile, unsupervised methods face difficulties in achieving high accuracy, particularly when addressing the semantic nuances of text and detecting implicit keyphrases[3]. Other traditional methods also have limitations since they only rely on statistical features without considering the semantic relationships between phrases and words in the document. This limitation

Received: 03-01-2025 | Accepted: 14-03-2025 | Published Online: 20-03-2025

causes the method to fail to capture the true meaning of documents, especially in complex and context-rich texts.

One of the methods that combine graph and topic approaches is Topic-guided graph attention networks [1]. Topic-guided graph Attention Networks (TgGAT) combine topic information such as topics of the document and context with graph structure, thus enhancing the model's ability to extract keyphrases more contextually and accurately. The research indicates that adding topic modeling leads to a significant decrease in performance, with an average F1@15 decrease of 5.46% which directly demonstrates the importance of utilizing topic information for keyphrase extraction tasks.

Trained with large data sets making the use of pre-trained language models has several advantages in that they can create a contextual understanding of the data and are efficient in terms of data [2]. Pre-Trained Language Models have proven to be effective in enhancing various natural language processing tasks. Pre-trained language models aim to predict relationships between sentences by analyzing them as a whole[3]. In natural language processing, pre-training models are based on language modeling tasks that aim to predict the next token or word [4]. Bidirectional Encoder Representations from Transformers (BERT) is one of the popular pre-trained language models that uses bidirectional self-attention[5]. Unlike other pre-trained language models that process text unidirectionally, BERT is designed to process bidirectionally, allowing BERT to understand word context based on the surrounding words[3], the architecture of BERT can be seen in Figure 1. Another Popular and advanced Pre-trained Language Model is RoBERTa (Robustly Optimized BERT Pre-Training Approach). The research findings indicate that during the training process, RoBERTa achieved an accuracy level 10% higher than BERT [2], allowing the system to attain better performance results by utilizing a Pre-trained Language Model with higher accuracy. Because it utilizes a pre-trained Language Model that has a higher accuracy level. Based on the explanation above, a Keyphrase Extraction system will be developed using the RoBERTa as Pre-trained Language Model and TgGAT (Topic Guided Graph Attention Networks).

Advancements in keyphrase extraction using Pre-Trained Language Models (PLMs), such as BERT and RoBERTa, have demonstrated impressive results, surpassing traditional methods. Furthermore, the use of PLM embeddings gives these models an edge over other embedding-based approaches, enhancing their performance in keyphrase extraction tasks [6][7]. In recent years NLP tasks developed significantly after implementing and utilizing Pre-Trained Language Models [8] especially Pre-trained Language Model BERT (BiDirectional Encoder Representations From Transformers) [9] and Pre-Trained Language Models RoBERTa (Robustly Optimized BERT Pretraining Approach) [10] because of their ability to understand the context between words in two directions. BERT and RoBERTA understand word context based on the words around it in a BiDirectional (two-direction) way [9].

The use of pre-trained languages such as BERT and RoBERTa in keyword extraction tasks has been carried out previously, such as the use of the pre-trained language model BERT-AKG [11] with generative methods showing significant improvements and outperforming all baselines. Apart from that, research on unsupervised keyphrase extraction using LMRank [12] and SIFRank [13] with embedding from the pre-trained language model BERT also succeeded in showing that the proposed method outperformed all other approaches[11]. Other research related to keyword extraction on small datasets uses a classification-based approach, because of limited data the pre-trained language model RoBERTa is used and shows that the research outperforms the baseline which proves that it is effective even on small datasets [14] Hierarchical graph representation ranking model [15] research in Graph-BERT field shows that the use of pre-trained and graph-based language models improves the system performances. Other research on keyword extraction systems with a graph-based approach shows that the graph-based approach in unsupervised tasks has the best performance because this method builds a word into a graph based on the occurrence of words in the document and ranks the words [16]-[18].

Despite advancements in keyphrase extraction, most existing studies apply either PLMs or graph-based models independently, without exploring their combined potential. PLMs, such as BERT and RoBERTa, excel in capturing contextual information but struggle with topic coherence and implicit keyphrase identification. Meanwhile, graph-based models effectively structure word relationships but lack deep contextual understanding.

To address these challenges, this research explores the integration of a graph-based approach (TgGAT) with pre-trained language models (BERT and RoBERTa) to improve keyphrase extraction. By leveraging both semantic understanding from PLMs and structured word relationships from TgGAT.s. By integrating Topic-Guided Graph Attention Networks (TgGAT) with PLMs, this study aims to bridge these gaps, combining PLMs with graph-based approaches like TgGAT presents a promising direction for improving keyphrase extraction performance. By modeling topic-keyphrase relationships explicitly, TgGAT can enhance the extraction process, addressing the shortcomings of PLMs in capturing implicit keyphrases.

Several studies have explored keyphrase extraction using either graph-based models or PLMs, but these approaches have primarily been applied to English-language datasets. Most existing research does not integrate both techniques, missing the opportunity to leverage their complementary strengths. This study is

the first to introduce a combined graph-based model (TgGAT) and PLMs for keyphrase extraction in the Indonesian language. While previous studies have used graph-based methods or PLMs independently, our approach integrates both techniques to enhance performance, setting a new benchmark for Indonesian keyphrase extraction.

The structure of this paper is arranged as follows. Section 2 represents the methodology used in this research, Section 3 represents the experimental result, and Section 4 represents the conclusions of this research.

2. Research Methods

The methods in this research are developed for keyphrase extraction from raw text documents, comparing the pre-trained language models BERT and RoBERTa. In this study, *raw documents* refer to unstructured text sources, primarily research paper abstracts, which encapsulate the essence of full-length studies in a condensed format. These documents often contain complex sentence structures, domain-specific terminology, and implicit relationships between key terms, making automated keyphrase extraction a challenging task that requires systematic processing.

The keyphrase extraction process begins with a pre-processing stage, where raw text undergoes normalization techniques such as tokenization, lowercasing, punctuation removal, and stopword elimination to enhance data consistency. After that, the data undergoes processing by pre-trained language models BERT or RoBERTa, which generate contextualized vector representations of words. These representations enable the system to capture semantic relationships and contextual dependencies within the document, improving the identification of meaningful keyphrases.

Subsequently, to further refine the extraction process, neural topic modeling is employed to identify the main themes present in the document. This step clusters related terms, providing insights into the document's topic structure. The extracted topic information is then utilized to construct an anchor-aware graph, representing the interconnections between words and phrases within the text. This graph is modeled using Topic-Guided Graph Attention Networks (TgGAT), which assigns weight distributions to words based on their significance in both local and global contexts. Phrases with the highest weights become salient nodes that encapsulate the essence of the text and form the system's output.

As a result, the system effectively identifies and extracts the most relevant keywords from the document, facilitating the indexing of the document in search systems. This enables users to find documents based on keywords that align with the topics or issues of their interest or search criteria. The architecture model is visualized in Figure 1.

Based on Figure 1, the framework begins with text preprocessing of the document, including case folding, cleaning, tokenization, pos-tagging, and noun-phrase chunking. Before proceeding to the noun phrase chunking stage, the document undergoes a POS-tagging process. This step involves identifying the part-of-speech category for each token in the text, and assigning labels such as nouns, verbs, and adjectives. POS-tagging provides crucial information that enhances the accuracy of noun phrase identification during the chunking phase. Following this, the noun phrase chunking process extracts only the noun phrases from the text, simplifying the representation and focusing on key components.

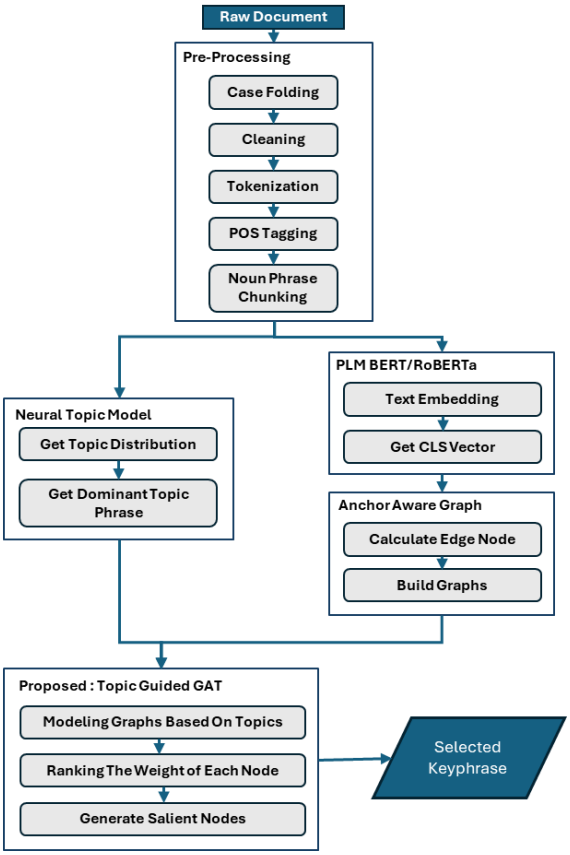

Figure 1. Model Framework

After that, the extracted noun phrases are then processed in two parallel branches: pre-trained language model (BERT/RoBERTa & anchor-aware graph) and Neural Topic Modeling. The extracted noun phrases are fed into a pre-trained language model (BERT or RoBERTa) to generate text embeddings. The CLS vector from the embeddings is obtained to represent the document's overall context. Once noun phrases are extracted, a pre-trained language model, RoBERTa, is employed to analyze the relationships between words within the text. RoBERTa enhances the system's ability to comprehend the context and extract meaning with greater depth and precision. An anchor-aware graph is then constructed by calculating edge nodes and defining relationships between words, ensuring that semantically related terms

are properly connected. This graph highlights the connections between words while emphasizing those with significant topical relevance, creating a structured representation of the relationships within the text.

On the other branch, the results of noun phrase chunking are processed with neural topic modeling utilizing Latent Dirichlet Allocation (LDA) to obtain the topic distribution for each phrase. This module takes the extracted noun phrases as input and identifies the main topics relevant to the document. By doing so, the system gains insights into the core focus of the text and isolates the most pertinent information. The dominant topic phrase is then extracted, providing additional contextual information that helps refine the keyphrase extraction process.

Next, the outputs from both branches are combined and processed in the Topic-Guided Graph Attention Network (TgGAT) stage, and the weighting of each phrase is ranked. From the ranking results, the top node, which represents the selected keywords, is obtained and identified the most salient nodes, which represent the final selected keyphrases This model is responsible for filtering and ranking keywords that are relevant to the document. By guiding the graph attention based on the topic, this model can extract the most relevant and informative keywords. Subsequently, an analysis will be conducted on both the author's keywords (golden keyphrase) and the keywords generated by the system.

2.1 *Pre-Trained Language Models BERT &RoBERTa*

One key distinction between RoBERTa and BERT lies in their pre-training methodologies. BERT utilizes the masked language modeling (MLM) approach, in which certain words within a sentence are masked, requiring the model to predict those hidden words or phrases.

From Figure 2, BERT involves a transformer encoder with a multi-layer transformer encoder that works to process and understand text, where each layer has two sub-layers, namely the multi-head self-attention mechanism and a feed-forward neural network [19]. In this study, the BERT model used is IndoBERT, which is a pre-trained language model trained on the Indonesian language corpus and follows the BERT architecture[20]. In its process, IndoBERT has the same architecture as it uses a BERT-base with a transformer mechanism that functions to learn relationships between words [21].

In contrast, RoBERTa enhances the MLM technique with several optimizations and eliminates the Next Sentence Prediction (NSP) step.

RoBERTa also involves longer training durations and using larger datasets, which contributes to its enhanced performance in natural language processing tasks., it employs more sophisticated and diverse data augmentation techniques, such as sentence order shuffling and token randomization. These techniques enable RoBERTa to better capture the context and relationships between words and phrases within a sentence. Although RoBERTa outperforms BERT in certain NLP tasks, both models share similar architectures and foundational principles[9].

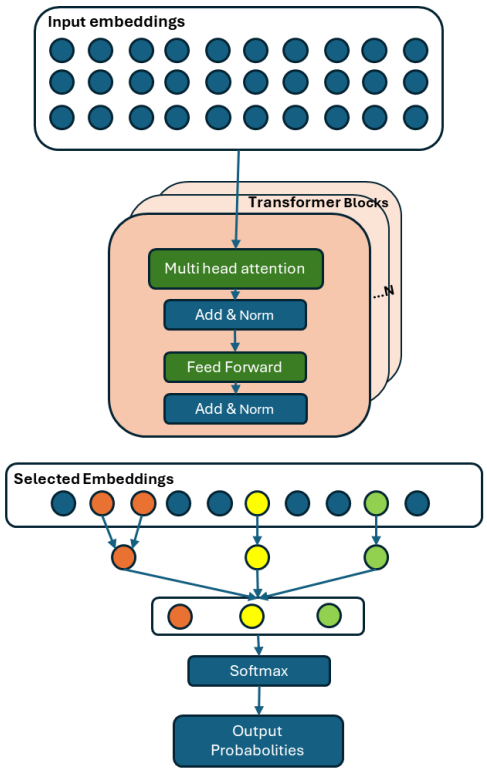

Figure 2. Sentence Transformers BERT Architecture [9]

RoBERTa is now considered as one of the top-performing NLP models and is extensively applied in various domains that demand advanced and accurate natural language processing. It is classified as a deep learning-based implementation in the field of natural language processing. [10]. Figure 3 illustrates the architecture of RoBERTa in a sentence-pair classification task.

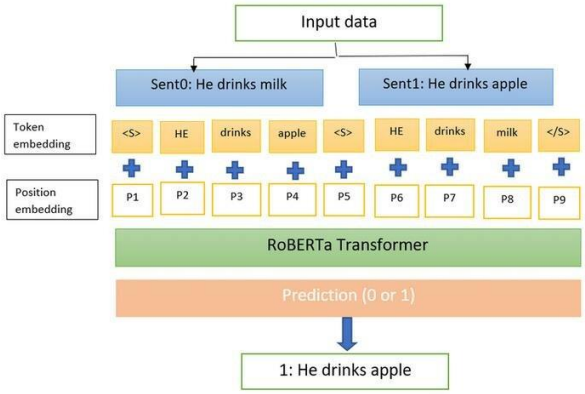

Figure 3. Sentence Transformers RoBERTa Architecture [22]

Based from Figure 3, the model takes two input sentences and encodes them using token and position embeddings before passing them through the RoBERTa transformer layers. Each input sentence is tokenized, and special tokens (<s>, </s>) are added to mark sentence boundaries. The token embeddings represent

individual words or subwords, while position embeddings help preserve the word order. These embeddings are summed and fed into the RoBERTa transformer, which processes the input using multiple self-attention layers. The final hidden states are then used for prediction, where the model outputs a binary classification (0 or 1) to determine the relationship between the sentences. This architecture is commonly used for tasks such as natural language inference, semantic similarity detection, and paraphrase identification.

The use of anchor-aware graphs leverages information from these anchors to identify relevant keyphrases. During modeling with anchor-aware graphs, phrases are extracted using BERT, which provides informative phrase features [1]. These graphs transform representations into a graph structure, where relationships between nodes are represented by edges. The anchor-aware graph calculates the similarity of edges between nodes representing the global context using Cosine Similarity, detailed in Equation 1.

$$o_{ij} = \frac{(r_i)^T . r_j}{\|r_i\| . \|r_j\|} \quad (1)$$

o_{ij} is the variable measuring the strength of the edge between nodes, r_i and r_j are the representations of phrases i and j within the graph, $\|r_i\|$ and $\|r_j\|$ are numerical values indicating the strength or prominence of the phrases. This method allows for the quantitative assessment of phrase relationships within the graph, facilitating the identification of key phrases that are most relevant and contextually significant to the overall document or text. Graph attention networks themselves are a type of graph neural network that considers the importance of each neighboring node and assigns weighting factors to each node connection [16]. TgGAT employs P as independent attention heads, where each attention head acts as a unit within the attention mechanism. These attention heads are capable of focusing on different data within the document, identifying diverse patterns and relationships. The formulation of independent attention heads is described in Equations 2 and 3:

$$a_{ij}^p = Softmax\,(LeakyReLU\,(a^T\left[W^p r_i; W^p r_j\right]; W^p r^M)) \quad (2)$$

$$r_i' = \left[\sigma\left(\sum_j a_{ij}^1 W^1 r_j\right); \ldots \sigma\left(\sum_j a_{ij}^P W^P r_j\right)\right], \quad (3)$$

The node representation, representing a phrase or keyword, is generated by segmenting the token representations from the encoder, which encapsulates the key information of the tokens forming the keyword. LeakyReLU (Leaky Rectified Linear Unit), a non-linear function, serves as the activation function. The non-linear activation function is applied to a value to generate a new value within a specific range. are attention coefficients that calculate the influence of one node on another. and are parameters, and are the representation of the topic embedded from the NTM.

3. Results and Discussions

The testing in this research was conducted on three sample data sets to evaluate the F1-score, comparing the system's extracted keyphrases with the author-generated keyphrases (golden keyphrases). The software testing utilized a test dataset of 100 scientific journal abstracts and titles. This testing served as an evaluation of the keyphrase extraction results achieved using the pre-trained language model BERT and the topic-guided graph attention networks. The evaluation metrics used are precision, recall, and F-score to evaluate a method from various statistical-based perspectives because it analyzes the performance of a method by calculating the proportion of the number of various key phrases, such as the number of key phrases extracted, correct keyphrases, incorrect keyphrases, and manually assigned keyphrases.

Table 1. Average Evaluation Matrix of 5 Keywords from 3 Example Documents Using BERT

Data	TP	FP	FN	TN	P	R	Fscore
1	2	3	4	23	0.4	0.34	0.36
2	2	3	3	46	0.4	0.34	0.40
3	2	3	2	38	0.4	0.5	0.44
Average					0.4	0.41	0.4

Table 1 which is tested on three samples with 5 keyphrases showed an average precision of 0.4, recall of 0.41, and F1-score of 0.4, indicating a balanced evaluation matrix. Meanwhile, testing on three samples with 10 keyphrases indicated an increase in the system's ability to identify true positives (TP) in two of the tests. However, there was a high incidence of false positives (FP), suggesting the system often misidentified non-keyphrases as keyphrases. Table 2 details the evaluation matrix values and the average for 10 keyphrases from each test data.

Table 2. Average Evaluation Matrix of 5 Keywords from 3 Example Documents Using RoBERTa

Data	TP	FP	FN	TN	P	R	Fscore
1	1	23	14	2	0.06	0.3	0.11
2	1	35	14	4	0.06	0.2	0.1
3	2	27	13	2	0.13	0.5	0.21
Average					0.08	0.33	0.14

Table 2 represents the Confusion Matrix, precision, recall, f-score and accuracy values for 3 sample datasets with the top 5 keyphrase parameters. Based on Table 2, the test results for the top 5 keyphrases were obtained with an average precision value of 0.08, recall of 0.33, and f-score of 0.14

Table 3. Average Evaluation Matrix of 10 Keywords from 3 Example Documents Using BERT

Data	TP	FP	FN	TN	P	R	Fscore
1	3	7	3	17	0.3	0.5	0.37
2	3	7	2	40	0.3	0.6	0.40
3	2	8	2	31	0.2	0.5	0.28
Average					0.26	0.53	0.35

High incidence of false positives (FP) in table 3 is attributed to the larger number of keyphrases extracted by the system, leading to more mismatches with the

golden keyphrases. The average F1 score for 10 keyphrases was lower compared to the score for 5 keyphrases. Otherwise, testing with 15 keyphrases showed improved performance in identifying true positives, but, as before, the false positives increased due to many system-generated keyphrases not matching the golden keyphrases. This increase in false positives led to lower precision compared to tests with 5 and 10 keyphrases. However, the F1-score for 15 keyphrases was lower than for 5 and 10 keyphrases. Table 3 displays the evaluation matrix values and averages for the 15 keyphrase tests.

Table 4. Average Evaluation Matrix of 10 Keywords from 3 Example Documents Using RoBERTa

Data	TP	FP	FN	TN	P	R	Fscore
1	1	31	9	4	0.1	0.2	0.133
2	1	16	9	5	0.1	0.16	0.125
3	1	29	9	2	0.1	0.33	0.154
				Average	0.1	0.23	0.13

Table 4 represents the Confusion Matrix, precision, recall, f-score and accuracy values for 3 sample datasets with the top 10 keyphrase parameters. Based on Table 4, the test results for the top 5 keyphrases were obtained with an average precision value of 0.1, recall of 0.23, and f-score of 0.13

Table 5. Average Evaluation Matrix of 15 Keywords from 3 Example Documents Using BERT

Data	TP	FP	FN	TN	P	R	Fscore
1	4	11	2	11	0.27	0.67	0.38
2	3	12	2	35	0.2	0.6	0.3
3	2	13	2	46	0.13	0.5	0.21
Average					0.20	0.59	0.30

Table 6. Average Evaluation Matrix of 15 Keywords from 3 Example Documents Using RoBERTa

Data	TP	FP	FN	TN	P	R	Fscore
1	2	12	13	4	0.13	0.33	0.19
2	1	24	14	2	0.06	0.33	0.11
3	1	27	14	2	0.06	0.33	0.11
				Average	0.08	0.33	0.13

However, similar to before, in Table 3 the false positive values decrease due to the large number of keywords generated by the system that do not match the golden keyphrase. With the increasing false positive values, the precision values decrease even more when compared to testing with 5 keywords and 10 keywords. Conversely, the recall values increase because the false negative values, or keywords not identified by the system, decrease. The F1-score results in testing with 15 keywords show a decline compared to testing with 5 keywords and 10 keywords.

Table 6 represents the Confusion Matrix, precision, recall, f-score and accuracy values for 3 sample datasets with the 15 top keyphrase parameters. Based on Table 6, the test results for the top 15 keyphrases were obtained with an average precision value of 0.08, recall of 0.33, and f-score of 0.13

Each dataset uniquely influences the frequency of true positives, false positives, and false negatives, and the precision, recall, and F1-score values on each model. The F1 scores from each test fluctuate, and results may vary with each run or execution of the program, leading to changes in precision, recall, and F1 score. Several factors contribute to the outcomes of the keyphrase extraction system, including the POS-Tagger model used for text preprocessing is not yet optimized, noun-phrase chunking still contains some word redundancies, stochastic phenomenon refers to how to graph attention networks initiate random weights causing variability in the system's results each time it is run, and the data used still contains noise and typographical errors. Therefore, before utilization or testing, it's necessary to check and clean the data to make it suitable for the system. Table 4 presents the evaluation matrix of the performance before data was cleaned and after data was cleaned.

Table 7. Evaluation Matrix Results Before and After Data Cleaning Using BERT

Before Cleaning			
Precision	Recall	F1-Score	Highest F1-Score
0.44	0.057	0.048	0.75
After Cleaning			
Precision	Recall	F1-Score	Highest F1-Score
0.058	0.070	0.062	0.5

From Table 7, it's evident that the F1-score improved after cleaning the dataset. Additionally, a test was conducted on 20 pre-selected and 10 pre-selected data samples. The selected 20 and 10 data samples contained fewer foreign terms and special characters such as %, (, and). The abundance of unrecognized foreign terms by the tagger made the phrase selection process less optimal. Table 5 shows the evaluation matrix results for these data samples.

Table 8. Comparison of Testing on Amount of Data Using BERT

Amount Data	Precision	Recall	F1-Score
100	0.058	0.070	0.062
20	0.090	0.115	0.097
10	0.160	0.223	0.184

From Table 8, it's observed that testing with 20 and 10 selected data samples resulted in better outcomes compared to processing the entire dataset (100 data samples) that were not pre-selected. The selected and tested data demonstrate the significant impact of data optimization on the keyphrase extraction system's process. In this study, the used data were not yet optimized affecting the keyphrase extraction results. Reviewing these factors reveals that data quality and noun-phrase chunking significantly influence the system's output. Table 6 displays the average F1-score for the entire 100 datasets with 5 extracted keyphrases.

Table 9. Average Evaluation Matrix for the Entire Dataset Using RoBERTa

Amount Data	Average Precision	Recall	F1-Score
100	0.026	0.03	0.027
20	0.0529	0.21455	0.0846
10	0.0462	0.076	0.076

Table 9 shows the results of keyword extraction using the pre-trained language model Roberta with precision 0.026, recall 0.03, dan f1-score 0.027 for 100 data and produces lower precision, recall and f1score values than the pre-trained language model BERT. When the dataset size is reduced to 20 samples, the model shows an increase in performance, with precision reaching 0.0529, recall 0.21455, and F1-score 0.0846. Similarly, for a dataset of 10 samples, the model records a precision of 0.0462, recall 0.076, and F1-score 0.076.

These results indicate that the RoBERTa model exhibits lower precision, recall, and F1-score compared to the BERT pre-trained language model, suggesting that BERT may be more effective for this specific keyword extraction task. The difference in performance could be attributed to variations in training objectives, tokenization, or the pre-training corpus used in both models.

4. Conclusions

Keyphrase Extraction (KPE) is a natural language processing (NLP) task that involves extracting a keyword related to the main topic discussed in a document. The increasing volume of information and documents leads people to spend a significant amount of time searching for relevant information based on keywords. Therefore, a system that can automatically extract keywords is needed to make the document search process more effective. Keyphrase extraction plays various important roles in the field of natural language processing. Keywords can be used in information retrieval systems, document exploration, facilitating quick document reading through visualizing important phrases, and organizing documents. Previous keyphrase extraction methods have shown limitations in modeling topic knowledge or the ability to model global information. They often focus too much on localized feature modeling, resulting in the extraction of homogeneous and less varied key phrases. This research employs an automatic keyphrase extraction approach based on a graph-based method. The study utilizes pre-trained language models, specifically BERT and RoBERTA, and Topic-guided graph attention networks. The use of pre-trained language models creates contextual understanding of data and efficiency in terms of data. Additionally, the usage of topic-guided graph attention networks serves to model a graph with topic information learned from NTM (neural topic modeling). This helps in modeling global context based on the graph-attention network architecture to enhance the coverage of key phrases and capture the context of a document. The research successfully created a keyphrase extraction system with precision, recall, and f1-score values of 0.058, 0.070, and 0.062, respectively, on 100 test data consisting of abstracts, titles, and keywords from scientific journal publications for Pre-Trained Language Models BERT and precision 0.026 , recall 0.03, f1-score 0.027 for Pre-trained Language Models RoBERTa. In this research it can be seen that pre-trained language models BERT are more effective and superior in terms of keyword extraction to pre-trained language models. Testing was also conducted on three samples of test data for 5 keywords, 10 keywords, and 15 keywords. The best results were achieved when the system produced 5 keywords.

Acknowledgements

This research was supported by DIPA of Public Service Agency of Sriwijaya University 2024. SP DIPA 023.17.2.677515/2024, On November 24, 2023. In accordance with the Dean's Decree Number: 0154/UN9.FIK/TU.SK/2024 On July 16, 2024.

References

[1] X. Zhu, Y. Lou, J. Zhao, W. Gao, and H. Deng, "Generative non-autoregressive unsupervised keyphrase extraction with neural topic modeling," *Eng. Appl. Artif. Intell.*, vol. 120, p. 105934, Apr. 2023, doi: 10.1016/j.engappai.2023.105934.

[2] D. Wu, W. Ahmad, and K.-W. Chang, "Pre-trained Language Models for Keyphrase Generation: A Thorough Empirical Study," Sep. 2022. doi: 10.48550/arXiv.2212.10233.

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2019, pp. 4171–4186. doi: 10.18653/v1/N19-1423.

[4] H. Wang, J. Li, H. Wu, E. Hovy, and Y. Sun, "Pre-Trained Language Models and Their Applications," *Engineering*, vol. 25, pp. 51–65, Jun. 2023, doi: 10.1016/j.eng.2022.04.024.

[5] S. Singla, "Comparative Analysis of Transformer Based Pre-Trained NLP Models," *Int. J. Comput. Sci. Eng.*, vol. 8, pp. 40–44, Dec. 2020, doi: 10.26438/ijcse/v8i11.4044.

[6] M. Song, Y. Feng, and L. Jing, "A Survey on Recent Advances in Keyphrase Extraction from Pre-trained Language Models," in *Findings of the Association for Computational Linguistics: EACL 2023*, A. Vlachos and I. Augenstein, Eds., Dubrovnik, Croatia: Association for Computational Linguistics, May 2023, pp. 2153–2164. doi: 10.18653/v1/2023.findings-eacl.161.

[7] N. Giarelis and N. Karacapilidis, "Deep learning and embeddings-based approaches for keyphrase extraction: a literature review," *Knowl. Inf. Syst.*, Jul. 2024, doi: 10.1007/s10115-024-02164-w.

[8] M. Song, Y. Feng, and L. Jing, "A Survey on Recent Advances in Keyphrase Extraction from Pre-trained Language Models," in *Findings of the Association for Computational Linguistics: EACL 2023*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2023, pp. 2153–2164. doi: 10.18653/v1/2023.findings-eacl.161.

[9] R. Devika, S. Vairavasundaram, C. S. J. Mahenthar, V. Varadarajan, and K. Kotecha, "A Deep Learning Model Based on BERT and Sentence Transformer for Semantic Keyphrase Extraction on Big Social Data," *IEEE Access*, vol. 9, pp. 165252–165261, 2021, doi: 10.1109/ACCESS.2021.3133651.

[10] Y. Liu *et al.*, "RoBERTa: A Robustly Optimized BERT Pretraining Approach," *CoRR*, vol. abs/1907.11692, 2019, [Online]. Available: http://arxiv.org/abs/1907.11692

[11] R. Liu, Z. Lin, and W. Wang, "Addressing Extraction and Generation Separately: Keyphrase Prediction With Pre-Trained Language Models," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 29, pp. 3180–3191, 2021, doi: 10.1109/TASLP.2021.3120587.

[12] N. Giarelis and N. Karacapilidis, "LMRank: Utilizing Pre-Trained Language Models and Dependency Parsing for Keyphrase Extraction," *IEEE Access*, vol. 11, pp. 71459–71471, 2023, doi: 10.1109/ACCESS.2023.3294716.

[13] Y. Sun, H. Qiu, Y. Zheng, Z. Wang, and C. Zhang, "SIFRank: A New Baseline for Unsupervised Keyphrase Extraction Based

on Pre-Trained Language Model," *IEEE Access*, vol. 8, pp. 10896–10906, 2020, doi: 10.1109/ACCESS.2020.2965087.

[14] S.-E. Kim, J.-B. Lee, G.-M. Park, S.-M. Sohn, and S.-B. Park, "RoBERTa-Based Keyword Extraction from Small Number of Korean Documents," *Electronics*, vol. 12, p. 4560, Sep. 2023, doi: 10.3390/electronics12224560.

[15] Z. Zhang, X. Liang, Y. Zuo, and C. Lin, "Improving unsupervised keyphrase extraction by modeling hierarchical multi-granularity features," *Inf. Process. Manag.*, vol. 60, no. 4, p. 103356, Jul. 2023, doi: 10.1016/j.ipm.2023.103356.

[16] Y. Ying, T. Qingping, X. Qinzheng, Z. Ping, and L. Panpan, "A Graph-based Approach of Automatic Keyphrase Extraction," *Procedia Comput. Sci.*, vol. 107, pp. 248–255, 2017, doi: 10.1016/j.procs.2017.03.087.

[17] M. Garg and M. Kumar, "KEST: A graph-based keyphrase extraction technique for tweets summarization using Markov Decision Process," *Expert Syst. Appl.*, vol. 209, p. 118110, Dec. 2022, doi: 10.1016/j.eswa.2022.118110.

[18] A. Mishra *et al.*, *GraphEx: A Graph-based Extraction Method for Advertiser Keyphrase Recommendation*. 2024.

[19] A. Vaswani *et al.*, "Attention Is All You Need," *CoRR*, vol. abs/1706.03762, 2017, [Online]. Available: http://arxiv.org/abs/1706.03762

[20] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained LanguageModel for Indonesian NLP," *CoRR*, vol. abs/2011.00677, 2020, [Online]. Available: https://arxiv.org/abs/2011.00677

[21] I. R. Hidayat and W. Maharani, "General Depression Detection Analysis Using IndoBERT Method," *Int. J. Inf. Commun. Technol.*, vol. 8, no. 1, pp. 41–51, Aug. 2022, doi: 10.21108/ijoict.v8i1.634.

[22] H. Al-Jarrah, R. Al-Hamouri, and M. AL-Smadi, "HR@JUST Team at SemEval-2020 Task 4: The Impact of RoBERTa Transformer for Evaluation Common Sense Understanding," in *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, Stroudsburg, PA, USA: International Committee for Computational Linguistics, 2020, pp. 521–526. doi: 10.18653/v1/2020.semeval-1.64.

