Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871

Vol. 7, No. 2, April 2026, Page. 875-890
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5320

Comparative Analysis of GPT-2 Augmentation, ALBERT, and Similarity Measures for Cyberbullying Detection

Zidane Hidayat*[1], Hasan Dwi Cahyono[2], Fajar Muslim[3]

[1,2,3]Faculty of Information Technology and Data Science, Universitas Sebelas Maret, Indonesia

Email: [1]zidanehidayat.zh@student.uns.ac.id

Received : Sep 23, 2025; Revised : Dec 7, 2025; Accepted : Dec 14, 2025; Published : Apr 15, 2026

Abstract

The effectiveness of cyberbullying detection is influenced by the availability of sufficient, diverse, and contextually rich training data, which is often limited in low-resource languages such as Indonesian. To address dataset limitations, researchers have extensively explored data augmentation (DA) as a promising approach to improving model performance. DA generates new data instances by applying transformations to existing data, thereby increasing both dataset size and variability. Prior studies have demonstrated that applying Easy Data Augmentation (EDA) with Support Vector Machine (SVM) classification improved cyberbullying detection performance, even when it faced challenges in capturing semantic and contextual nuances. In this paper, we investigated Indonesian DA methods using the Transformer-based GPT-2 model. The augmented sentences were evaluated and filtered based on context, semantics, diversity, and novelty, with similarity measures such as Euclidean Distance (ED), Cosine Similarity (CS), Jaccard Similarity (JS), and BLEU Score (BLS) ensuring the quality of the augmentation. Furthermore, we compared text classification performance using both SVM and the Transformer-based ALBERT model. Experimental results revealed that incorporating similarity measures and GPT-2 as a DA method failed to improve cyberbullying detection performance, potentially due to the semantic drift introduced by GPT-2 and the inadequacy of similarity measures in capturing nuanced contextual information. However, we found that ALBERT outperformed SVM as a classification model, achieving average F1-scores of 91.77% and 91.72%, respectively. This study contributes to the informatics field by exploring the potential of Transformer-based augmentation and similarity evaluation in enhancing low-resource text classification, while acknowledging the limitations in data quality and model adaptation.

Keywords : *ALBERT, Cyberbullying, Data Augmentation, GPT-2, NLP, Similarity Measure*

This work is an open access article and licensed under a Creative Commons Attribution-Non Commercial 4.0 International License

1. INTRODUCTION

Cyberbullying is one of the most prevalent problems in today's interconnected world, affecting a wide range of individuals. These phenomena are understood as aggressive actions that are repeated with the intention of hurting and shaming the victim through social media or other digital communication technologies. Some of the actions that are considered cyberbullying are flaming, harassment, denigration, impersonation, outing, exclusion, trickery, and cyberstalking, all of which have the potential to cause severe damage to the victim's psychology and social life [1]. With fundamental characteristics such as intentionality, power imbalance, repetition, anonymity, and public or private scope, cyberbullying is increasingly common in everyday life. Because of that, NLP can be a potential solution in helping to prevent it.

Natural Language Processing (NLP) is a branch of Artificial Intelligence (AI) that aims to train computers to process and analyze large amounts of natural language data [2]. Text classification is a sub-branch of NLP that studies techniques for training AI to classify textual data based on predefined classes. The main objective of text classification is to help with text and document categorization based on their content. This technique has numerous practical applications, from sentiment analysis to content

Vol. 7, No. 2, April 2026, Page. 875-890
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5320

filtering. With the advancement within machine learning and deep learning technologies, text classification has become increasingly accurate and efficient in handling large amounts of data [3]. However, Machine Learning models' performance is highly dependent on the availability of data. Thus, limited data can limit the ability of NLP models [4].

Data Augmentation (DA) has become a promising approach to address the limitation of data availability [5], [6], [7]. DA increases the size of the training data instances by applying various transformation techniques to the actual dataset, generating new representative instances. Thereby, the training process of the Machine Learning models could be more optimal [6]. Generally, DA techniques for textual datasets are categorized into two main groups. The first category is a paraphrase-based technique, such as using a thesaurus [6], translation [8], or a Transformer [9]. The second category is adding noise to the text, such as swapping [10], deletion [4], insertion [11], and substitution [12]. Although DA is an established method within Computer Vision and Speech Recognition, its adoption in NLP is not yet widespread [13]. The traditional DA method requires complex computational power and is time-consuming, particularly on languages with insufficient resources for the augmentation process, such as language dictionaries or synonym databases for the selected datasets. Based on previous research, Easy Data Augmentation (EDA) has been proven to increase text classification performance for the topic of Indonesian-language cyberbullying [14]. Furthermore, not all augmentation methods can be implemented for every language because certain DA methods may generate grammatically or semantically incorrect instances [5], [15].

Using pre-trained Transformer models for DA can address the weaknesses of traditional DA methods [16]. The use of Transformer models for DA preserves the textual context and word dependencies of the original sentences [9], [17], [18]. However, it is also necessary to evaluate the quality of the generated instances to not only increase the number of cases but also preserve or increase the semantic quality and utility for NLP models. The quality mentioned earlier can be measured from various perspectives, such as context, semantics, diversity, and novelty [19]. Some similarity metrics can be used to quantitatively measure those aspects, such as ED [20] to measure the numerical distance between text representations, ED [21] to measure directional similarity of the text semantics based on their vector representations, JS [22] which compares text lexical overlap, and BLS [23] as an indicator of n-gram-based linguistic suitability.

Various DA techniques have been implemented for multiple languages, with the majority focusing on English cases [4], [9], [24]. DA methods have been proven to be effective in enhancing English computational training performance and have not been extensively explored in Indonesian language cases. In contrast, Indonesian itself is used by more than 250 million people, making it one of the largest languages in the world based on the number of speakers [25]. One of the challenges of Indonesian in the digital era is that its usage often does not comply with the standard Indonesian language rules [26]. This non-conformity creates unique complexity for DA in Indonesian cases. Indonesian has diverse morphological characteristics, such as affixation (prefixes, suffixes, and infixes) and reduplication, which are often not followed consistently in casual discussions on social media or in short messages.

Furthermore, the inclusion of local languages (such as Javanese and Sundanese) or foreign languages (such as English and Japanese) increases the complexity of automated text analysis in Indonesian. From an informatics perspective, Indonesian cyberbullying detection represents a typical low-resource NLP challenge where data scarcity, semantic drift, and inconsistent morphology reduce model reliability. Understanding how augmentation quality interacts with Transformer architectures is, therefore, essential for developing scalable moderation systems and advancing the scientific foundations of Indonesian NLP. Prior studies [14] have demonstrated that traditional augmentation can enhance the performance of cyberbullying detection. However, they have not explored the integration of a

Vol. 7, No. 2, April 2026, Page. 875-890
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5320

Transformer and a similarity measure to maintain semantic consistency during the augmentation process. Transformer models have been proven to be a promising method in this field [24], [27].

This research introduces a novel approach that integrates similarity measures into Transformer-based data augmentation, specifically for Indonesian cyberbullying detection, and evaluates its effectiveness using both classical (SVM) and advanced (ALBERT) models. The combination of a Transformer for augmentation with similarity measures has also shown promising results [28]. Therefore, this research aims to explore the performance of these augmentation methods in generating more diverse datasets that are still consistent semantically, as well as assessing their impact on enhancing the accuracy of text classification in Indonesian compared to traditional augmentation methods.

2. METHOD

The steps of this research follow the workflow outlined in Figure 1. Research began with data preprocessing to ensure consistency and quality and ended with performance evaluation. This research presents four scenarios: EDA augmentation and SVM, EDA augmentation with a similarity measure and SVM, GPT-2 augmentation with a similarity measure and SVM, and GPT-2 augmentation with a similarity measure and ALBERT text classification.

2.1. Data Collection, Preprocessing, and Cross Validation

The dataset can be obtained from https://github.com/rizalespe/Dataset-Sentimen-Analisis-Bahasa-Indonesia/ and has also been used in previous studies [14]. The dataset consists of 400 data points divided equally into two classes, 200 positives and 200 negatives. In this context, the negative class is data that is considered cyberbullying, while the positive class is non-cyberbullying data. Before the experiment was conducted, preprocessing was carried out on the dataset to ensure consistency. The preprocessing stages included case folding, language normalization, removal of stop words, stemming, and tokenization. A 10-fold cross-validation scheme was selected because it offers stable performance estimates in small datasets, reduces variance across folds, maximizes data utilization, and prevents overfitting, as demonstrated in low-resource Indonesian NLP [14].

2.2. Augmentation

Next, the dataset will be augmented using Easy Data Augmentation (EDA) and GPT-2. In a previous study, EDA has been proven to enhance text classification performance [14]. Besides, GPT-2 has promising results as a Transformer for text augmentation [29].

a. *Easy Data Augmentation* (EDA)

The EDA method in Indonesian was conducted using the EDA method from https://github.com/jasonwei20/eda_nlp, which was previously employed in [14]. The thesaurus used for Synonym Replacement and Random Insertion was https://github.com/victoriasovereigne/tesaurus. EDA works by randomly doing one of the different EDA operations for each of the augmentation processes. Those operations are:

- Synonym Replacement (SR): Where the operation will randomly choose *n* number of words from the text and replace them with their synonyms.
- Random Insertion (RI): Where the operation will randomly find a synonym from words inside the text that aren't stop words and insert it into a random position inside the text. Within this research, this operation is turned off based on the previous research parameter [14].
- Random Swap (RS*)*: Where the operation will swap the position of two words from the text.
- Random Deletion (RD): Where the operation will delete each word within the text with the probability of *p*.

Vol. 7, No. 2, April 2026, Page. 875-890
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5320

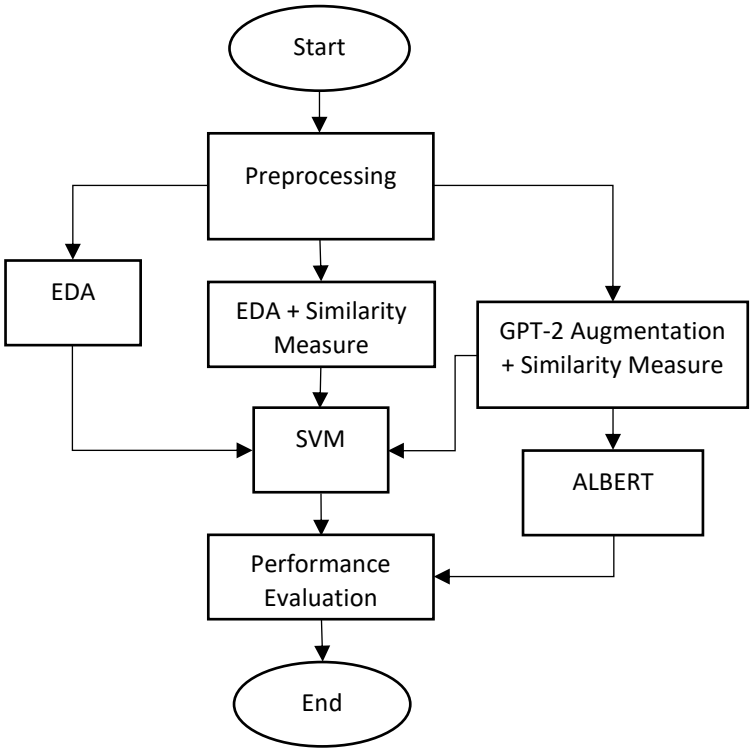

Figure 1. Research Workflow

EDA as a method has some limitations. This technique cannot capture the complexity of the text and language features such as irony and sarcasm. That limitation has the potential to muddy the context of the text and increase the noise within the data [29].

b. GPT2-Small-Indonesian-522M

GPT2-Small-Indonesian-522M is a language learning model based on GPT-2, which was developed for Indonesian Text. This model has 522 million parameters and is part of the smaller GPT-2 series, which is adapted and trained on Indonesian datasets. This repository can be accessed at pada 'cahya/gpt2-small-indonesian-522M' at *Hugging Face* [30]. The advantages of the GPT-2 model are the ability to understand context in text and augment the dataset with that context in mind [31]. In this study, the ratio of generated sentences during the augmentation process follows the parameter used in [14]. For each original sentence, we will generate 1, 2, 4, 8, 16, and 32 sentences across different scenarios.

2.3. Similarity Measure

During this part, the augmented dataset is filtered with four similarity metrics: ED, CS, JS, and BLS, to ensure the quality and relevance of the augmentation results. For each metric, the threshold is defined as the mean of the similarity scores of the entire augmented results, providing an adaptive and standardized cutoff. Only text with a similarity score higher than the threshold is retained to maintain the quality of the dataset. Following the procedure of the previous study [28], ED was treated using the same thresholding rule, where Euclidean values above the threshold were retained in a similar manner. The filtered dataset is saved and then used for evaluation, as shown in Table 1.

Table 1. Dataset Statistic with similarity measure

Initial Training Dataset	Augmentation				Similarity Measure		Total Training Dataset (Initial Dataset + Retained Dataset)
	Method	n_{aug}	α (for EDA)	Generated	Metric	Retained	

Vol. 7, No. 2, April 2026, Page. 875-890
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5320

360	EDA	1	0.05	360	BLS	142	502
					CS	43	403
					ED	314	674
					JS	108	468
			0.1	360	BLS	169	529
					CS	41	401
					ED	316	676
					JS	110	470
			0.2	360	BLS	211	571
					CS	105	465
					ED	252	612
					JS	135	495
			0.3	360	BLS	211	571
					CS	105	465
					ED	252	612
					JS	135	495
			0.4	360	BLS	218	578
					CS	108	468
					ED	249	609
					JS	122	482
			0.5	360	BLS	216	576
					CS	142	502
					ED	204	564
					JS	128	488
		2	0.05	720	BLS	278	638
					CS	99	459
					ED	620	980
					JS	204	564
			0.1	720	BLS	343	703
					CS	93	453
					ED	626	986
					JS	210	570
			0.2	720	BLS	411	771
					CS	154	514
					ED	565	925
					JS	285	645
			0.3	720	BLS	434	794
					CS	208	568
					ED	509	869
					JS	278	638
			0.4	720	BLS	433	793
					CS	239	599
					ED	478	838
					JS	248	608
			0.5	720	BLS	426	786
					CS	280	640
					ED	429	789
					JS	248	608
		4	0.05	1440	BLS	587	947
					CS	175	535
					ED	1264	1624
					JS	399	759
			0.1	1440	BLS	683	1043
					CS	210	570
					ED	1229	1589
					JS	444	804
			0.2	1440	BLS	812	1172
					CS	294	654
					ED	1145	1505
					JS	543	903
			0.3	1440	BLS	852	1212
					CS	428	788
					ED	1009	1369
					JS	559	919
			0.4	1440	BLS	850	1210

Vol. 7, No. 2, April 2026, Page. 875-890
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5320

	0.5	1440	CS	495	855
			ED	939	1299
			JS	446	806
			BLS	877	1237
			CS	560	920
			ED	854	1214
			JS	462	822
8	0.05	2880	BLS	1154	1514
			CS	343	703
			ED	2536	2896
			JS	831	1191
	0.1	2880	BLS	1399	1759
			CS	394	754
			ED	2485	2845
			JS	844	1204
	0.2	2880	BLS	1674	2034
			CS	547	907
			ED	2332	2692
			JS	1139	1499
	0.3	2880	BLS	1695	2055
			CS	786	1146
			ED	2093	2453
			JS	1119	1479
	0.4	2880	BLS	1740	2100
			CS	992	1352
			ED	1865	2225
			JS	953	1313
	0.5	2880	BLS	1714	2074
			CS	1129	1489
			ED	1703	2063
			JS	942	1302
16	0.05	5400	BLS	2169	2529
			CS	648	1008
			ED	4751	5111
			JS	1494	1854
	0.1	5400	BLS	2604	2964
			CS	712	1072
			ED	4687	5047
			JS	1681	2041
	0.2	5400	BLS	3127	3487
			CS	1080	1440
			ED	4319	4679
			JS	2117	2477
	0.3	5400	BLS	3176	3536
			CS	1535	1895
			ED	3858	4218
			JS	2098	2458
	0.4	5400	BLS	3225	3585
			CS	1863	2223
			ED	3505	3865
			JS	1792	2152
	0.5	5400	BLS	3251	3611
			CS	2105	2465
			ED	3147	3507
			JS	1778	2138
32	0.05	9720	BLS	3806	4166
			CS	1156	1516
			ED	8563	8923
			JS	2734	3094
	0.1	9720	BLS	4726	5086
			CS	1319	1679
			ED	8402	8762
			JS	2954	3314
	0.2	9720	BLS	5655	6015
			CS	1999	2359

Vol. 7, No. 2, April 2026, Page. 875-890
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5320

		0.3	9720	ED	7720	8080
				JS	3822	4182
				BLS	5719	6079
				CS	2763	3123
				ED	6944	7304
				JS	3758	4118
		0.4	9720	BLS	5810	6170
				CS	3356	3716
				ED	6311	6671
				JS	3255	3615
		0.5	9720	BLS	5837	6197
				CS	3760	4120
				ED	5757	6117
				JS	3229	3589
GPT-2	1	-	360	BLS	149	509
				CS	111	471
				ED	348	708
				JS	139	499
	2	-	720	BLS	303	663
				CS	225	585
				ED	699	1059
				JS	301	661
	4	-	1440	BLS	625	985
				CS	451	811
				ED	1399	1759
				JS	605	965
	8	-	2880	BLS	1213	1573
				CS	903	1263
				ED	2799	3159
				JS	1190	1550
	16	-	5760	BLS	2450	2810
				CS	1807	2167
				ED	5599	5959
				JS	2470	2830
	32	-	11520	BLS	4922	5282
				CS	3615	3975
				ED	11199	11559
				JS	4586	4946

a. BLEU Score (BLS)

The BLEU (Bilingual Evaluation Understudy) Score measures the quality of the text produced by an automatic translation system or text augmentation process. The BLS compares the n-*grams* of the produced text and the reference texts to determine the similarity. A high BLS indicates that the produced text is highly similar to the reference text.

The BLS is measured using this equation. First, count the mean geometric of the modified precision *n-gram* (p_n) using *n-grams* of length N where positive w_n is equal to one. Next, if c is the length of the candidate translation and r is the effective length of the reference corpus, Brevity Penalty (BP) is applied to provide a penalty c that is smaller than r as in Equations (1) and (2) [23].

$$BP = \begin{cases} 1 & if\ c > r \\ e^{(1-r/c)} & if\ c \leq r \end{cases} \quad (1)$$

$$\log BLEU = BP.\exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \quad (2)$$

b. Cosine Similarity (CS)

CS uses the cosine angle between two texts to measure their similarity. CS values range from -1 to 1, with a value of 1 indicating that the two texts are identically similar. CS is widely used in text

Vol. 7, No. 2, April 2026, Page. 875-890
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5320

classification and information retrieval because it can ignore differences in document length. In text modeling, documents are converted into vectors where the values in each dimension of the vector contain the frequency of occurrence of a word in the document [32]. If we assume the vectors of the two documents as *x* and *y*, the formula for CS can be written as equation (3).

$$\cos(x, y) = \frac{x.y}{\|x\|.\|y\|} \tag{3}$$

c. Euclidean Distance (ED)

ED measures the similarity between two points within a multidimensional space [20]. In the context of NLP, ED is used to measure the difference between the representation vectors of two texts. A smaller ED value shows that the texts are quite similar to each other [36]. The ED value or *d* between two points, *a* for text 1 and *b* for text 2, which represent the text vector in n dimensions, where each index *k* represents each dimension, is calculated using Equation (4).

$$d(a, b) = \sqrt{\sum_{k=1}^{n} (a_k - b_k)^2} \tag{4}$$

d. Jaccard Similarity (JS)

JS is assessed between two sets based on the ratio of the number of elements they share to the number of unique elements in the two sets [22]. In NLP, each document is represented as a set of unique words, where similarity is calculated based on the ratio of the number of words that appear in both documents (intersection) to the total number of unique words that appear in either document (union). This method is ideal for sparse data because it only considers the presence of a word (value 1) and ignores the much more dominant absence of a word (value 0) [32]. If we consider the two texts being compared as sets T_1 and T_2, with intersection denoted as ∩, and union denoted as ∪, then JS can be written as Equation (5).

$$J(T_1, T_2) = \frac{|T_1 \cap T_2|}{|T_1 \cup T_2|} \tag{5}$$

2.4. Text Classification

After undergoing the augmentation and filtering stages using similarity measures, the final dataset was utilized for text classification. This study employed two different classification algorithm approaches: Support Vector Machine (SVM) and ALBERT. SVM was chosen because it has been used in previous research[14]. Meanwhile, ALBERT is considered because it has superior semantic representation capabilities and text classification performance [27]. The performance of the classifications was evaluated based on a 10-fold cross-validation for training accuracy, testing accuracy, and F1-score, as reported in previous research [14].

a. Support Vector Machine

Support Vector Machine (SVM) is an algorithm that can handle linear and nonlinear classification tasks. In the nonlinear case, a kernel function projects the input data into a higher-dimensional space, allowing the identification of an optimal hyperplane that maximizes the separation margin between

Vol. 7, No. 2, April 2026, Page. 875-890
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5320

classes. This hyperplane serves as a decision boundary between two categories, typically denoted as class +1 and class −1, where each represents a distinct pattern [33]. In this study, SVM was implemented using the scikit-learn library. To determine the most appropriate parameters, a grid search method was used, which systematically evaluates all possible parameter combinations within a specified range.

b. ALBERT

ALBERT (A Lite BERT) is a pre-trained BERT derivative model developed by Google in 2020, which is a lighter model than BERT. ALBERT uses two parameter reduction strategies to overcome the main limitation in resizing pre-trained models. The first strategy is to use embedding parameterization, where the insertion matrix with an extensive vocabulary is split into two smaller matrices. This splitting facilitates the segregation of hidden layers in the data and also the size of the embedded vocabulary. Therefore, increasing the amount of hidden data becomes easier without increasing the value of the existing vocabulary embedding. The second strategy is to split the parameters at several levels [34].

2.5. Scenario Summary Table

Table *2* presents the list of scenarios implemented in this study. S1 represents the initial scenario based on [15] as the baseline for the experiments attempted. Next, we apply filtering using a similarity measure in S2 in conjunction with the application of EDA. After that, we introduce a generative augmentation method by substituting EDA with GPT-2 as the augmentation model. Finally, we further maximize the use of Transformers by also substituting SVM into ALBERT as the classifier model.

Table 2. Our Research Scenario

Scenario	Augmentation	Filtering	Classifier	Notes
S1	EDA	No	SVM	Lexical baseline
S2	EDA	Yes	SVM	EDA + semantic filtering
S3	GPT-2	Yes	SVM	Generative augmentation
S4	GPT-2	Yes	ALBERT	Full Transformer pipeline

3. RESULT

3.1. Research Environment

Experiments are conducted using a computer with the following specifications: Intel Core i7 Gen 12th processor. 32GB of memory, 1TB SSD storage, NVIDIA RTX 5060 Ti 16GB graphics card, and Windows 11 as the operating system. To maintain reproducibility, the program source code is developed in Python 3.11.11 and can be accessed from uns.id/comparative-cyberbullying.

3.2. Similarity Measure Implementation

For the first part, we implement similarity measures to evaluate and increase the data quality produced by EDA. During the augmentation process, every text produced is compared to the original and scored with the BLS, CS, ED, and JS. An example of the result of the augmentation is shown in Table 3. After that, the mean score of the entire dataset is calculated and used as a threshold, retaining only text with a score higher than the threshold, as shown in Table 4. The result of the augmentation is then used in the classification process, which utilizes SVM. The results can be seen in Table 5.

Table 3. Example of the Result of EDA with Its Similarity Score

Text	label	new_text	original_ embedding	new_ embedding	ES	CS	JS	BLS
bencong rombengmenjijikangay jg	0	bencong jg rombengmenjijikangay	0.036, 0.019, -0.017, -0.008,	0.036, 0.019, -0.017, -0.008, 0.019, 0.014, ...	0.000	1.000	1.000	1.000

Vol. 7, No. 2, April 2026, Page. 875-890
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5320

			0.019, 0.014, ...					
duh geulis kitu tinggal duuuhhhh hahaha jodoh allah kasih teeeh	1	duh geulis kitu tinggal duuuhhhh hahaha jodoh allah berahi teeeh	0.039, 0.026,-0.039,-0.016, 0.002, 0.014, …	0.039, 0.026,-0.039,-0.016, 0.002, 0.014, …	0.000	0.999	0.882	0.900
ulang sampet gak percaya klo asli bagus bgt	1	ulang sampet gak percaya klo asli elok bgt	0.031, 0.026, -0.031, -0.030, 0.002, 0.012, …	0.031, 0.026, -0.031, -0.030, 0.002, 0.012, …	0.000	0.999	1.000	0.880

Table 4. Example of the Dataset after Filtering

Text	Label
bencong rombengmenjijikangay jg	0
duh geulis kitu tinggal duuuhhhh hahaha jodoh allah kasih teeeh	1
duh geulis kitu tinggal duuuhhhh hahaha jodoh allah berahi teeeh	1
ulang sampet gak percaya klo asli bagus bgt	1
ulang sampet gak percaya klo asli elok bgt	1

Table 5. The Result of EDA, Similarity Measure, and SVM Method

n_{aug}	α	Similarity Measure	Average Similarity score	Training	Testing	
				Accuracy	Accuracy	F1 score
1	0.05	BLEU	0.933	86.45%	87.50%	87.50%
		Cosine	0.986	86.59%	87.50%	87.50%
		Euclidean	0.048	86.20%	87.50%	87.50%
		Jaccard	0.970	87.38%	87.50%	87.50%
	0.1	BLEU	0.928	88.48%	90.00%	90.00%
		Cosine	0.985	86.29%	87.50%	87.50%
		Euclidean	0.048	87.12%	90.00%	90.00%
		Jaccard	0.974	87.66%	90.00%	90.00%
	0.2	BLEU	0.879	86.06%	90.00%	90.00%
		Cosine	0.976	85.11%	90.00%	90.00%
		Euclidean	0.085	87.66%	87.50%	87.50%
		Jaccard	0.964	85.51%	90.00%	90.00%
	0.3	BLEU	0.836	86.33%	90.00%	90.00%
		Cosine	0.966	86.24%	87.50%	87.50%
		Euclidean	0.115	86.11%	90.00%	90.00%
		Jaccard	0.954	84.64%	90.00%	90.00%
	0.4	BLEU	0.778	86.15%	90.00%	90.00%
		Cosine	0.963	87.18%	87.50%	87.50%
		Euclidean	0.124	87.03%	90.00%	90.00%
		Jaccard	0.945	86.06%	90.00%	90.00%
	0.5	BLEU	0.753	85.25%	87.50%	87.50%
		Cosine	0.951	85.86%	90.00%	90.00%
		Euclidean	0.170	85.64%	87.50%	87.50%
		Jaccard	0.935	83.60%	87.50%	87.50%
...	...	...	...	...	...	...
16	0.05	BLEU	0.934	88.45%	90.00%	90.00%
		Cosine	0.986	87.50%	90.00%	90.00%
		Euclidean	0.047	88.00%	87.50%	87.50%
		Jaccard	0.973	88.40%	90.00%	90.00%
	0.1	BLEU	0.925	89.71%	90.00%	90.00%
		Cosine	0.925	88.71%	90.00%	90.00%
		Euclidean	0.051	86.56%	90.00%	90.00%
		Jaccard	0.972	87.55%	90.00%	90.00%
	0.2	BLEU	0.880	85.00%	90.00%	90.00%
		Cosine	0.978	86.94%	87.50%	87.50%
		Euclidean	0.077	86.66%	90.00%	90.00%
		Jaccard	0.964	86.43%	90.00%	90.00%
	0.3	BLEU	0.829	**85.52%**	**95.00%**	**95.00%**
		Cosine	0.968	86.54%	87.50%	87.50%
		Euclidean	0.111	86.54%	90.00%	90.00%
		Jaccard	0.953	86.13%	87.50%	87.50%

Vol. 7, No. 2, April 2026, Page. 875-890
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5320

	0.4	BLEU	0.788	84.38%	90.00%	90.00%
		Cosine	0.961	85.24%	85.00%	85.00%
		Euclidean	0.136	86.52%	92.50%	92.50%
		Jaccard	0.945	85.83%	82.50%	82.50%
	0.5	BLEU	0.753	83.11%	87.50%	87.50%
		Cosine	0.953	83.28%	85.00%	85.00%
		Euclidean	0.162	83.83%	90.00%	90.00%
		Jaccard	0.939	85.31%	87.50%	80.00%
...	...	...	...	...	...	...
AVG				86.28%	88.98%	88.92%

3.3. Change the augmentation method to GPT-2

In the next section, we modify the augmentation method to utilize a Transformer. In this research, the Transformer that we use is fine-tuned *GPT-2-Small-Indonesian-522M*. An example of the result can be seen in Table 6. Next, the mean similarity score from the entire dataset is calculated and used as a threshold like before, as shown in Table 7. The augmented dataset was then classified with the SVM method. The results are presented in Figure 2.

3.4. Changing the classification method with ALBERT

In the next part, we used ALBERT as the substitute text classification model. Since the GPT-2 augmentations are the same as those in the previous part, the dataset was used again. For fine-tuning, ALBERT was trained for 10 epochs with a batch size of 16 using the Adam optimizer, a linear learning-rate scheduler with zero warm-up steps, a maximum sequence length of 512 tokens, and gradient clipping with a maximum norm of 1.0. The performance result from that text classification can be seen in Figure 3.

3.5. Result Summary

EDA consistently outperformed GPT-2 when paired with SVM because the lexical variations introduced by EDA align more closely with the informal patterns of the Indonesian language. GPT-2, however, tended to generate semantically drifting output, even after filtering, which reduced the classical model's stability. CS in general has consistently been the best similarity metric for evaluating augmentation results compared to other similarity metric. ALBERT achieved the highest overall performance because its contextual modeling compensates for the structural noise present in augmented data. Across all experiments, performance variance remained within ±2%, indicating high experimental stability. A variance analysis showed that all models exhibited low fold-to-fold deviation (<2%), confirming the reliability of observed patterns. Although formal significance testing (e.g., t-tests) was not applied due to computational constraints, the consistent performance trend across multiple scenarios provides strong empirical support for the findings.

Table 6. Example of the Result of GPT-2 Augmentation with Its Similarity Score

Text	label	new_text	original_ embedding	new_ embedding	ED	CS	JS	BLS
bencong rombengmenjijikangay jg	0	bencong rombengmenjijikangay jg bngt langgeng pisah otak as	0.036, 0.019, -0.017, -0.008, 0.019, 0.014, ...	0.036, 0.019, -0.017, -0.008, 0.019, 0.014, ...	0.000	1.000	0.700	0.690
duh geulis kitu tinggal duuuhhhh hahaha jodoh allah kasih teeeh	1	duh geulis kitu tinggal duuuhhhh hahaha	0.039, 0.026,- 0.039,-	0.039, 0.026,- 0.038, -	0.000	0.999	0.833	0.710

Vol. 7, No. 2, April 2026, Page. 875-890
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5320

		jodoh allah kasih teeehh kereet bunda asih liat	0.016, 0.002, 0.014, …	0.016, 0.002, 0.014, …			
ulang sampet gak percaya klo asli bagus bgt	1	ulang sampet gak percaya klo asli bagus bgtdenger udh yaaa kaya	0.031, 0.026, -0.031, -0.030, 0.002, 0.012, …	0.031, 0.026, -0.031, -0.030, 0.002, 0.012, …	0.000	0.999	0.857	0.970

Table 7. Example of the Dataset after Filtering

Text	label
bencong rombengmenjijikangay jg	0
bencong rombengmenjijikangay jg bngt langgeng pisah otak as	0
duh geulis kitu tinggal duuuhhhh hahaha jodoh allah kasih teeeh	1
duh geulis kitu tinggal duuuhhhh hahaha jodoh allah kasih teeehh kereet bunda asih liat	1
ulang sampet gak percaya klo asli bagus bgt	1

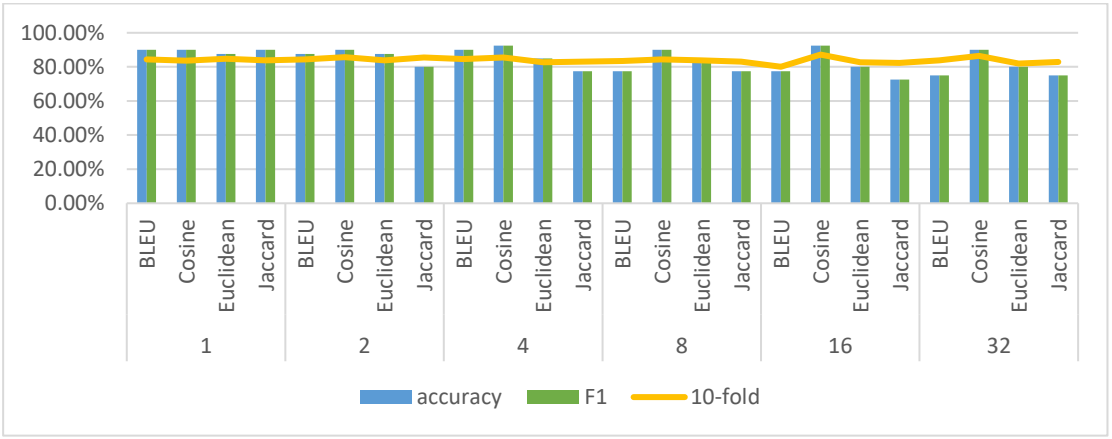

Figure 2. The Graph of the Result of GPT-2 Augmentation, Similarity Measure, and SVM Method

Vol. 7, No. 2, April 2026, Page. 875-890
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5320

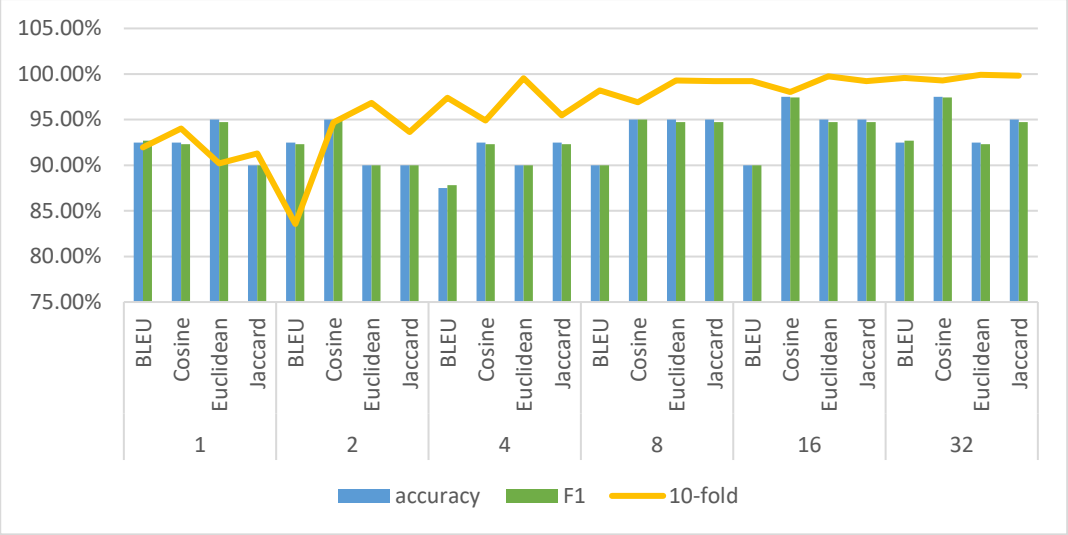

Figure 3. The Graph of Result of the GPT-2 Augmentation, Similarity Measure, and ALBERT

4. DISCUSSION

Table 8. Comparison of the Average Performance of Each Method

Model	Average Across All Settings		
	Training	Testing	
	Accuracy	Accuracy	F1 Score
SVM + EDA [14]	87.13%	90.76%	-
SVM + EDA (ours)	86.10%	88.89%	88.89%
SVM + EDA + Similarity Measure	86.28%	88.98%	88.92%
SVM + GPT-2 Augmentation + Similarity Measure	83.95%	84.06%	84.06%
ALBERT + GPT-2 Augmentation + Similarity Measure	**96.33%**	**92.92%**	**92.83%**

Table 8 provides a comparison of the various methods used. We compared the best and average results for each method's settings. The average results were compared to assess the overall performance of the methods. Based on these results, the implementation of the Similarity Measure and the substitution of the augmentation method from EDA to the GPT-2 for augmentation did not significantly improve text classification performance. The implementation of the Similarity Measure and GPT-2 yielded similar results to those of the previous study. Even then, we found that CS is among the most consistently beneficial similarity metrics to use for augmentation, generally yielding a 2.50% increase in performance compared to other similarity metrics we use.

Meanwhile, the ALBERT model for classification appears to have successfully improved text classification performance. The classification results using the combination of GPT-2 augmentation, Similarity Measure, and ALBERT achieved a training performance improvement of 96.33%, with a testing accuracy of approximately 92.92%, an F1-score of roughly 92.83. This study is limited by the dataset size, the mismatch between GPT-2 pretraining data and informal Indonesian, and the computational demands of fine-tuning the Transformer.

Compared to [14], who used only EDA, our findings show that generative augmentation must be paired with semantic filtering to prevent noisy training signals. Similarly, prior Indonesian NLP studies demonstrate that Transformer-based models generally outperform classical approaches when contextual semantics are essential. Our results reinforce this pattern, showing that ALBERT provides superior robustness even when trained on imperfect synthetic data [1].

Vol. 7, No. 2, April 2026, Page. 875-890
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5320

5. CONCLUSION

Based on the research results, several conclusions can be drawn. First, the application of similarity measures in the data augmentation process fails to improve text classification performance. Second, the use of the GPT-2-based augmentation method fails to produce more accurate results compared to simple augmentation using EDA. Third, ALBERT benefits more from the augmentation process in general, where increasing the amount of data has been shown to improve model performance consistently. Overall, this study indicates that the quality of augmentation through similarity measures and the use of the GPT-2 model is not yet fully optimal for the Indonesian text classification task, with further research can be attempted with better-suited Transformer models. However, the availability of large amounts of data remains a significant factor in enhancing model performance, particularly for deep learning-based approaches.

CONFLICT OF INTEREST

The authors declare that there is no conflict of interest between the authors or with the research object in this paper.

ACKNOWLEDGEMENT

Our deepest gratitude is extended to RG Dike at Universitas Sebelas Maret for the support and assistance provided under the HGR A 2025 research scheme no. 371/UN27.22/PT.01.03/2025.

REFERENSI

[1] Y. E. Riany and F. Utami, "Cyberbullying Perpetration among Adolescents in Indonesia: The Role of Fathering and Peer Attachment," *Int Journal of Bullying Prevention*, May 2023, doi: 10.1007/s42380-023-00165-x.

[2] U. Kamath, J. Liu, and J. Whitaker, *Deep Learning for NLP and Speech Recognition*. Cham: Springer International Publishing, 2019. doi: 10.1007/978-3-030-14596-5.

[3] C. Manning, P. Raghavan, and H. Schuetze, *Introduction to Information Retrieval*. 2009.

[4] J. Wei and K. Zou, "EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks," Aug. 25, 2019, *arXiv*: arXiv:1901.11196. doi: 10.48550/arXiv.1901.11196.

[5] J.-P. Corbeil and H. A. Ghadivel, "BET: A Backtranslation Approach for Easy Data Augmentation in Transformer-based Paraphrase Identification Context," Sept. 25, 2020, *arXiv*: arXiv:2009.12452. doi: 10.48550/arXiv.2009.12452.

[6] X. Dai and H. Adel, "An Analysis of Simple Data Augmentation for Named Entity Recognition," in *Proceedings of the 28th International Conference on Computational Linguistics*, D. Scott, N. Bel, and C. Zong, Eds., Barcelona, Spain (Online): International Committee on Computational Linguistics, Dec. 2020, pp. 3861–3867. doi: 10.18653/v1/2020.coling-main.343.

[7] G. Daval-Frerot and Y. Weis, "WMD at SemEval-2020 Tasks 7 and 11: Assessing Humor and Propaganda Using Unsupervised Data Augmentation," in *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, Barcelona (online): International Committee for Computational Linguistics, 2020, pp. 1865–1874. doi: 10.18653/v1/2020.semeval-1.246.

[8] A. Fabbri *et al.*, "Improving Zero and Few-Shot Abstractive Summarization with Intermediate Fine-tuning and Data Augmentation," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, and Y. Zhou, Eds., Online: Association for Computational Linguistics, June 2021, pp. 704–717. doi: 10.18653/v1/2021.naacl-main.57.

[9] Y. Hou, S. Chen, W. Che, C. Chen, and T. Liu, "C2C-GenDA: Cluster-to-Cluster Generation for Data Augmentation of Slot Filling," *AAAI*, vol. 35, no. 14, pp. 13027–13035, May 2021, doi: 10.1609/aaai.v35i14.17540.

[10] C. Rastogi, N. Mofid, and F.-I. Hsiao, "Can We Achieve More with Less? Exploring Data Augmentation for Toxic Comment Classification," July 02, 2020, *arXiv*: arXiv:2007.00875. doi: 10.48550/arXiv.2007.00875.

[11] G. Yan, Y. Li, S. Zhang, and Z. Chen, "Data Augmentation for Deep Learning of Judgment Documents," in *Intelligence Science and Big Data Engineering. Big Data and Machine Learning*, vol. 11936, Z. Cui, J. Pan, S. Zhang, L. Xiao, and J. Yang, Eds., in Lecture Notes in Computer Science, vol. 11936. , Cham: Springer International Publishing, 2019, pp. 232–242. doi: 10.1007/978-3-030-36204-1_19.

[12] Q. Xie, Z. Dai, E. Hovy, M.-T. Luong, and Q. V. Le, "Unsupervised data augmentation for consistency training," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, in NIPS '20. Red Hook, NY, USA: Curran Associates Inc., 2020.

[13] C. Shorten, T. M. Khoshgoftaar, and B. Furht, "Text Data Augmentation for Deep Learning," *Journal of Big Data*, vol. 8, no. 1, p. 101, July 2021, doi: 10.1186/s40537-021-00492-0.

[14] A. Wirawan, H. D. Cahyono, and Winarno, "Easy Data Augmentation in Sentiment Analysis of Cyberbullying," in *2023 6th International Conference on Information and Communications Technology (ICOIACT)*, 2023, pp. 443–447. doi: 10.1109/ICOIACT59844.2023.10455817.

[15] S. Y. Feng *et al.*, "A Survey of Data Augmentation Approaches for NLP," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds., Online: Association for Computational Linguistics, Aug. 2021, pp. 968–988. doi: 10.18653/v1/2021.findings-acl.84.

[16] A. Vaswani *et al.*, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, in NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, pp. 6000–6010.

[17] T. Kober, J. Weeds, L. Bertolini, and D. Weir, "Data Augmentation for Hypernymy Detection," in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, P. Merlo, J. Tiedemann, and R. Tsarfaty, Eds., Online: Association for Computational Linguistics, Apr. 2021, pp. 1034–1048. doi: 10.18653/v1/2021.eacl-main.89.

[18] K. Li, C. Chen, X. Quan, Q. Ling, and Y. Song, "Conditional Augmentation for Aspect Term Extraction via Masked Sequence-to-Sequence Generation," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds., Online: Association for Computational Linguistics, July 2020, pp. 7056–7066. doi: 10.18653/v1/2020.acl-main.631.

[19] A. Celikyilmaz, E. Clark, and J. Gao, "Evaluation of Text Generation: A Survey," May 18, 2021, *arXiv*: arXiv:2006.14799. doi: 10.48550/arXiv.2006.14799.

[20] R. Mussabayev, "Optimizing Euclidean Distance Computation," *Mathematics*, vol. 12, no. 23, p. 3787, Nov. 2024, doi: 10.3390/math12233787.

[21] J. Zobel and A. Moffat, "Exploring the similarity space," *SIGIR Forum*, vol. 32, no. 1, pp. 18–34, Apr. 1998, doi: 10.1145/281250.281256.

[22] G. Travieso, A. Benatti, and L. da F. Costa, "An Analytical Approach to the Jaccard Similarity Index," Oct. 21, 2024, *arXiv*: arXiv:2410.16436. doi: 10.48550/arXiv.2410.16436.

[23] R. Bawden, B. Zhang, L. Yankovskaya, A. Tättar, and M. Post, "A Study in Improving BLEU Reference Coverage with Diverse Automatic Paraphrasing," in *Findings of the Association for Computational Linguistics: EMNLP 2020*, T. Cohn, Y. He, and Y. Liu, Eds., Online: Association for Computational Linguistics, Nov. 2020, pp. 918–932. doi: 10.18653/v1/2020.findings-emnlp.82.

[24] J. Li, X. Zhang, and X. Zhou, "ALBERT-Based Self-Ensemble Model With Semisupervised Learning and Data Augmentation for Clinical Semantic Textual Similarity Calculation: Algorithm Validation Study," *JMIR Med Inform*, vol. 9, no. 1, p. e23086, Jan. 2021, doi: 10.2196/23086.

[25] "Demographic Statistics Indonesia (Results of Population Census 2020)." BPS-Statistics Indonesia, Jan. 31, 2025. Accessed: May 20, 2025. [Online]. Available:

Vol. 7, No. 2, April 2026, Page. 875-890
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5320

https://www.bps.go.id/en/publication/2025/01/31/29a40174e02f20a7a31b5bc3/demographic-statistics-indonesia--results-of-population-census-2020-.html

[26] L. Sundary and F. Fauzah, "Studi Analisis Perkembangan Bahasa Indonesia di Era Digital," *Innovative*, vol. 4, no. 3, pp. 11295–11303, June 2024, doi: 10.31004/innovative.v4i3.11633.

[27] S. F. N. Azizah, H. D. Cahyono, S. W. Sihwi, and W. Widiarto, "Performance Analysis of Transformer Based Models (BERT, ALBERT, and RoBERTa) in Fake News Detection," in *2023 6th International Conference on Information and Communications Technology (ICOIACT)*, 2023, pp. 425–430. doi: 10.1109/ICOIACT59844.2023.10455849.

[28] D. Refai, S. Abu-Soud, and M. J. Abdel-Rahman, "Data Augmentation Using Transformers and Similarity Measures for Improving Arabic Text Classification," *IEEE Access*, vol. 11, pp. 132516–132531, 2023, doi: 10.1109/ACCESS.2023.3336311.

[29] V. Maslej-Krešňáková, M. Sarnovský, and J. Jacková, "Use of Data Augmentation Techniques in Detection of Antisocial Behavior Using Deep Learning Methods," *Future Internet*, vol. 14, no. 9, p. 260, Aug. 2022, doi: 10.3390/fi14090260.

[30] N. A. Ranggianto, D. Purwitasari, C. Fatichah, and R. W. Sholikah, "Abstractive and Extractive Approaches for Summarizing Multi-document Travel Reviews," *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 7, no. 6, pp. 1464–1475, Dec. 2023, doi: 10.29207/resti.v7i6.5170.

[31] F. Sufi, "Generative Pre-Trained Transformer (GPT) in Research: A Systematic Review on Data Augmentation," *Information*, vol. 15, no. 2, p. 99, Feb. 2024, doi: 10.3390/info15020099.

[32] A. W. Qurashi, V. Holmes, and A. P. Johnson, "Document Processing: Methods for Semantic Text Similarity Analysis," in *2020 International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*, Novi Sad, Serbia: IEEE, Aug. 2020, pp. 1–6. doi: 10.1109/INISTA49547.2020.9194665.

[33] D. A. Pisner and D. M. Schnyer, "Support vector machine," in *Machine Learning*, Elsevier, 2020, pp. 101–121. doi: 10.1016/B978-0-12-815739-8.00006-7.

[34] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations," Feb. 09, 2020, *arXiv*: arXiv:1909.11942. doi: 10.48550/arXiv.1909.11942.