Sentiment Analysis of the Kampus Merdeka Program on Twitter Using Support Vector Machine and a Feature Extraction Comparison: TF-IDF vs. FastText

Lasmedi Afuan[1,*], Nurul Hidayat[2], Nofiyati[3], Mohamad Faris As'ad[4]

[1,2,3,4]*Department of Informatics, Engineering Faculty, Universitas Jenderal Soedirman, Indonesia*

(Received: July 19, 2024; Revised: August 13, 2024; Accepted: September 11, 2024; Available online: October 15, 2024)

Abstract

The Kampus Merdeka program, launched by the Indonesian Ministry of Education, Culture, Research, and Technology in 2020, aims to enhance students' skills through hands-on work experience. Considering the rising significance of social media, particularly Twitter, in gauging public opinion, this research focuses on analyzing the sentiment towards the Kampus Merdeka program. The primary objective is to classify the sentiments expressed in tweets related to the program and compare two feature extraction techniques—TF-IDF and FastText—to identify the best approach for transforming text data into numerical vectors. The sentiment classification model was built using the Support Vector Machine (SVM) algorithm, a machine learning technique known for its accuracy in text classification. A total of 16,730 tweets were collected and analyzed, yielding an accuracy of 73% for FastText and 72% for TF-IDF. Results show that FastText is more effective in capturing semantic relationships, leading to higher accuracy in sentiment classification. Findings indicate that the public sentiment towards the Kampus Merdeka program is predominantly positive (60.7%), with negative and neutral sentiments at 33.5% and 5.8%, respectively. The success of the FastText method underscores the importance of advanced feature extraction techniques in text classification. The novelty of this research lies in its use of FastText for educational policy evaluation, providing a new perspective on using sentiment analysis to assess public perception of educational programs.

Keywords: Sentiment Analysis, FastText, Kampus Merdeka, Support Vector Machine, TF-IDF, Twitter

1. Introduction

The transformation of higher education in Indonesia has become a strategic agenda to develop superior human resources that are globally competitive [1], [2]. One of the key initiatives launched by the Ministry of Education, Culture, Research, and Technology (Kemendikbudristek) in 2020 is the Kampus Merdeka program [3]. This program offers students opportunities to gain practical work experience through internships, independent studies, and teaching. Since its launch, the program has garnered significant public attention, especially among students and educators, who have voiced their opinions on social media platforms such as Twitter. These varied reactions present an opportunity to analyze public sentiment to better understand the perceptions surrounding the Kampus Merdeka program.

However, the issue arises from the lack of structured, measurable understanding of how the public, especially on Twitter, perceives the Kampus Merdeka program. Twitter, being one of the most popular social media platforms in Indonesia, provides a public space for open expression [4]. Nevertheless, transforming thousands, or even millions, of tweets into interpretable data requires appropriate analytical approaches. This is where sentiment analysis becomes crucial, enabling the categorization of public opinions into positive, negative, or neutral sentiments.

This research is conducted because of the scarcity of prior studies specifically addressing public sentiment toward the Kampus Merdeka program using text mining and machine learning approaches. Although previous studies have applied sentiment analysis to various other issues, there is a gap in comprehensive research that leverages this technology to measure public perceptions of educational policies in Indonesia, particularly this program. This knowledge gap must be filled to provide insights for policymakers in assessing the effectiveness and impact of the Kampus Merdeka

***Corresponding author: Lasmedi Afuan (lasmedi.afuan@unsoed.ac.id)**
DOI: https://doi.org/10.47738/jads.v5i4.436

ISSN 2723-6471

initiative. The main problem this research aims to solve is how to develop a sentiment analysis model that accurately classifies public opinions based on Twitter data. By utilizing machine learning methods, specifically the Support Vector Machine (SVM), and comparing two different feature extraction techniques—TF-IDF and FastText—this research seeks to determine the most effective method for understanding public sentiment toward the program.

Addressing this issue is critical because public perception is one of the key indicators of a successful public policy. If Kampus Merdeka is well-received, it can be further developed to provide greater benefits for students and the workforce. Conversely, if the program receives negative sentiment, urgent evaluation and revision are necessary. Thus, sentiment analysis becomes an important tool in data-driven policy evaluation.

To address the problem, this study will adopt a machine learning-based approach, which has proven effective in text classification tasks. Using SVM as the primary algorithm and comparing the performance of two feature extraction techniques (TF-IDF and FastText), this research will produce a model that is both accurate and efficient in analyzing large volumes of text data. SVM was chosen due to its high classification accuracy and robustness when dealing with textual data [5], [6], [7], [8], [9].

This research is conducted due to the need for a comprehensive understanding of public perceptions regarding national education policies. The findings of this study are expected to provide valuable input to Kemendikbudristek and other stakeholders regarding the success of Kampus Merdeka in achieving its goals. With measurable sentiment data, policymakers can make better decisions regarding the future development of the program.

The primary contribution of this research is filling the gap in the literature on sentiment analysis in the field of education policy. Although many previous studies have explored the implementation of machine learning for sentiment analysis in other sectors, few have focused on education policy in Indonesia. This study provides a fresh and important perspective on evaluating policy success through modern, data-driven analytical approaches.

The state of the art in this research lies in the use of FastText and TF-IDF for feature extraction in sentiment analysis. FastText is a deep learning-based embedding technique that allows for better word representation, particularly for handling uncommon words [10], [11], [12], [13]. Meanwhile, TF-IDF is a well-established technique for weighting words in text documents [14], [15], [16]. By comparing these two methods in the context of sentiment analysis, this research offers innovation in selecting the best feature extraction method to improve model accuracy.

The innovation proposed in this study is the use of a SVM approach combined with a performance comparison between FastText and TF-IDF. This research not only focuses on developing a sentiment analysis model but also aims to identify the optimal feature extraction technique for analyzing public sentiment in the context of education policy in Indonesia. Based on the research problem, two research questions are formulated: (1) What is the public sentiment toward the Kampus Merdeka program on Twitter? and (2) Which feature extraction method—TF-IDF or FastText—is more effective for sentiment classification using SVM? The main objective of this research is to develop an SVM-based sentiment analysis model that accurately classifies public opinions about Kampus Merdeka on Twitter. This study also compares the effectiveness of two feature extraction techniques, TF-IDF and FastText, to determine the best method for sentiment classification in this context. A machine learning-based methodology is employed for text data analysis. The data used are tweets related to Kampus Merdeka, collected from Twitter, and processed through multiple preprocessing stages before being analyzed using the SVM model. The research evaluates the model's performance based on accuracy, precision, recall, and F1-score metrics. The expected outcome is a sentiment analysis model that can accurately identify public perceptions and provide meaningful input for improving the Kampus Merdeka policy.

2. Methodology

In this study, the methodology consists of several stages, as shown in figure 1.

 ISSN 2723-6471

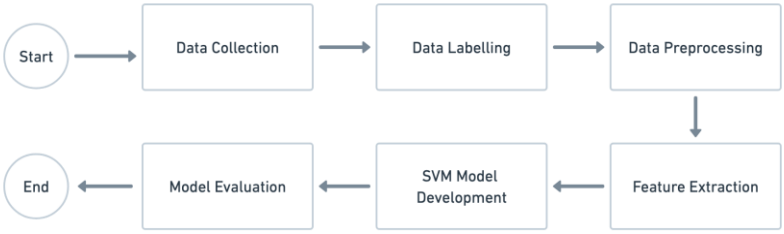

Figure 1. Research Methodology

The steps taken in the research methodology in figure 1 can be explained as follows.

2.1. Data Collection

The data used in this study consists of tweets from Twitter users containing the keywords "kampus merdeka" and "MBKM," which were gathered using the SNScraper module. This module allows comprehensive data scraping for specific timeframes, ensuring that the tweets accurately reflect public sentiment. The collected data is divided into two primary datasets: the training dataset and the testing dataset. The training dataset is used to develop the sentiment classification model, while the testing dataset is used to evaluate its performance. A total of 16,730 data points were gathered, and for consistency, 60% of the data (10,038 data points) were randomly selected for the training set, while the remaining 40% (6,692 data points) were used for testing. This split ratio ensures that the training dataset is large enough to train the model comprehensively while maintaining a sufficient amount of testing data to validate the model. The division was performed using a stratified sampling method to ensure that the sentiment distribution (positive, neutral, and negative) is balanced across both the training and testing datasets. This approach minimizes bias and ensures that the model's performance can be accurately assessed on unseen data. Additionally, unlabeled tweets were also included to simulate real-world scenarios where sentiment is not predetermined.

2.2. Data Labelling

After the data collection process, the data labeling stage was conducted to categorize the tweets into three sentiment categories: negative, positive, and neutral. To ensure the consistency and reliability of the labeling process, a set of predefined criteria and guidelines were established. These guidelines clearly defined what constitutes a negative, positive, or neutral sentiment based on the content and tone of the tweets. The manual labeling process involved three human annotators with backgrounds in data science and natural language processing. Each tweet was independently labeled by these annotators to minimize subjective interpretation. To ensure labeling consistency, the Cohen's Kappa coefficient was calculated to measure inter-annotator agreement. A Kappa score of above 0.8 indicated a strong agreement among the annotators, ensuring high reliability in the labeling process. For tweets that were ambiguous or contained mixed sentiments, a consensus-based approach was adopted. In such cases, annotators discussed the sentiment context and reached an agreement on the most appropriate label. For example, if a tweet expressed both criticism and praise, the annotators considered the primary intent and context to determine whether the overall sentiment leaned more toward positive, negative, or neutral. This detailed approach ensured that the labeled dataset was not only accurate but also credible, providing a strong foundation for training the sentiment classification model.

2.3. Data Preprocessing

Data preprocessing is a crucial stage in the sentiment analysis process, significantly impacting model accuracy and performance. Effective preprocessing ensures that the data is clean, standardized, and simplified, thereby enabling the machine learning model to focus on meaningful patterns in the text. In this research, several preprocessing steps were implemented to enhance the quality of the input data. The mechanism of the preprocessing steps that will be performed can be seen in figure 2.

 ISSN 2723-6471

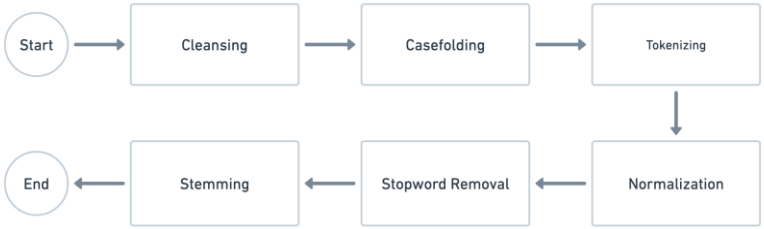

Figure 2. Preprocessing Data steps

The data preprocessing stages include several key steps: cleansing, case folding, tokenizing, normalization, stopword removal, and stemming. Cleansing involves removing unwanted characters, such as punctuation marks, special symbols, URLs, and hashtags to reduce noise and improve focus on relevant words. Case folding converts all text to lowercase to ensure uniformity and prevent the model from treating words in different cases (e.g., "Kampus" vs. "kampus") as separate features. Tokenizing splits sentences into individual words or tokens, enabling the model to capture the structure and context of sentences more effectively. Normalization converts non-standard words into their standard forms (e.g., “gak” to “tidak”), which is particularly useful in informal settings like social media. Stopword removal eliminates common words (e.g., “and,” “or,” “the”) that do not contribute to sentiment, reducing dimensionality and helping the model focus on sentiment-bearing terms. Finally, stemming reduces words to their root forms (e.g., “running” to “run”), minimizing redundancy and enhancing training efficiency. Together, these preprocessing steps contribute to improved accuracy, precision, recall, and F1-score by ensuring consistency and eliminating irrelevant variations, as demonstrated in the model evaluation section of this research. In the text mining process, there is word weighting, which aims to assign a value or weight to terms found in a document. This process is called feature extraction. Feature extraction in the context of text mining is the process of transforming words or text into a numerical representation that can later be used by machine learning algorithms to perform various analyses, such as classification, categorization, clustering, or prediction. In this study, two feature extraction methods will be used: TF-IDF and FastText.

2.3.1. TF-IDF

Term weighting is a process of assigning weights to words so that sentiment analysis results can be optimized. One type of term weighting is TF-IDF [17]. Term Frequency (tf(w,d)) is considered to have a proportional significance based on its total occurrences in a text or document. Inverse Document Frequency (IDF) is a token weighting method used to monitor the appearance of tokens in a set of texts. TF-IDF functions to calculate the frequency of word occurrences in a document, then converts those words into numerical values of 0. Equation (1) shows how to calculate tf, where tf indicates the number of occurrences of a term in a document.

$$tf_{td} = f_{td} \tag{1}$$

Equation (2) represents the calculation for IDF, where N is the total number of documents, and df is the number of documents containing the term t.

$$Idf = \log \frac{N}{df_t} \tag{2}$$

After obtaining the tf and idf values, the TFIDF value will then be calculated using equation (3)

$$W_{t,d} = tf_{t,d} \; X \; idf_{t,d} X \; \log \frac{N}{df_t} \tag{3}$$

Note (3): $W_{t,d}$ = TF-IDF weight; $tf_{t,d}$ = Word frequency count; $idf_{t,d}$ = Inverse document frequency count for each word; df_t = Document frequency count for each word; N = Total number of documents

The TF-IDF values obtained from Equation 3 will be normalized using L2 Normalization. The equation can be seen in Equation (4).

 ISSN 2723-6471

$$X_{(i,j)} = \frac{tfidf_{(i,j)}}{\sqrt{\sum_{j=j0}^{jn} tfidf^2(i,j)}} \quad (4)$$

$tfidf_{(i,j)}$ is the TF-IDF value of the word to be normalized, then $\sum_{j=j0}^{jn} tfidf^2(i,j)$ is the value of the Euclidean distance from the matrix where the document $tfidf_{(i,j)}$ comes from.

The result of word weighting using TF-IDF is the product of the TF and IDF values, which will produce a smaller weight if the word frequently appears in every document in the collection. Conversely, the TF-IDF weight will be larger if the word rarely appears in each document in the collection.

2.3.2. Fasttext

FastText is a type of word embedding method and a development of the Word2Vec method, where character n-grams are also used in word representation. FastText is an open-source project developed by the Facebook Research Lab team as an effective and fast method for word vectorization and text classification. The difference from Word2Vec, as previously explained, is that FastText works by involving subwords. For example, the word "kampus" will be broken down into <k, ka, am, mp, us>. In contrast, Word2Vec works with a single word "kampus." By representing words into a series of n-grams, unseen words in the corpus can be better represented because it is likely that some of the n-grams forming the word will appear in the n-grams found in the corpus. The illustration of how FastText works using the skip-gram architecture is shown in figure 3.

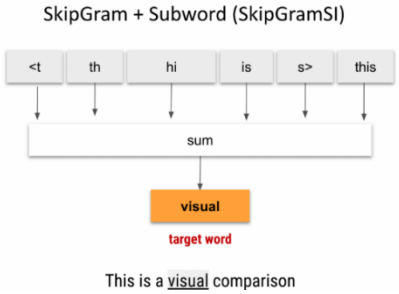

Figure 3. Illustration of FastText in skip-gram

2.4. Model Creation using SVM

In building the sentiment analysis model, the SVM method was chosen. The SVM algorithm is a type of supervised learning algorithm. SVM was selected as the primary algorithm in this study due to its strong theoretical foundations and proven effectiveness in handling high-dimensional feature spaces, such as those encountered in text classification tasks. Unlike other classifiers, SVM is particularly well-suited for problems where the number of features far exceeds the number of data points, a common scenario in text mining. The SVM method is a learning system that uses a hypothesis space consisting of linear functions in a high-dimensional feature space, trained using learning based on optimization theory [18]. SVM will attempt to find the best hyperplane (separator) that separates the data into two classes and maximizes the margin between these two classes. The hyperplane found by SVM is illustrated, as shown in figure 4.

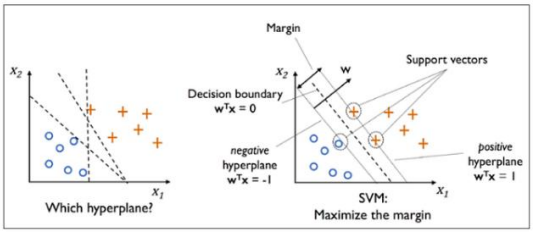

Figure 4. Hyperplane that separates between 2 classes

 ISSN 2723-6471

The hyperplane found by SVM is illustrated, as shown in figure 4, positioned in the middle between two classes. This means the distance between the hyperplane and the data points of the adjacent classes (marked with empty and positive circles) differs. In SVM, the outermost data points that are closest to the hyperplane are called support vectors. These support vectors are the most difficult to classify due to their near overlap with the other class. Given their critical nature, only these support vectors are considered when finding the most optimal hyperplane by SVM.

In some cases, data cannot be classified using a linear SVM method, so kernel functions are developed to classify the data in a nonlinear form. In general, commonly used kernel functions are the Linear, Polynomial, and Radial Basis Function (RBF) kernels. The steps in the SVM algorithm are as follows [19]. To begin, the model training dataset needs to be prepared. Once the dataset is ready, the appropriate kernel function should be selected for the Support Vector Machine (SVM) model. In this research, the Radial Basis Function (RBF) kernel is used, as illustrated in Equation (5).

$$K(X_1X_2) = \exp(-\gamma \left||X_1 - X_2|\right|^2) \tag{5}$$

After defining the kernel, the next step is to perform optimization. The goal of optimization is to minimize the margin value using the **Lagrange multiplier duality equation**, as given in Equation (6).

$$\max L_d(\alpha) = \sum_{i=1}^{n} a_i - \frac{1}{2} \sum_{i,j=1}^{n} a_i a_j y_i y_j K(x_i x_j) \tag{6}$$

This equation must satisfy the condition that: $a_i \geq 0$

$$\sum_{i=1}^{n} a_i y_i = 0$$

Once the optimization problem is solved, the resulting decision function (hyperplane) can be derived. The final output from the decision function equation will take the form provided in Equation (7).

$$\sum_{i \in SV} y_i \alpha_i K(x_i, x) + b, \tag{7}$$

In the case of multiclass classification, more than one hyperplane will be formed. There are several approaches that allow SVM to perform multiclass classification. Two common approaches for generalizing SVM to multiclass classification are One-vs-One (OvO) and One-vs-All (OvA). In the One-vs-One approach, the SVM will be trained for n classes. For example, if there are 3 classes, the model will be trained by distinguishing between class 1 and class 2, class 1 and class 3, class 2 and class 3, and so on. Predictions will then be made by taking the majority vote from all class comparisons. In contrast, in the One-vs-All approach, the model will be trained by comparing one class against the rest. For example, if there are 3 classes, training will be done with class 1 and not class 1, class 2 and not class 2, class 3 and not class 3. The model will then predict by taking the largest value from the hyperplane equation.

2.5. Model Evaluation

The model evaluation will later analyze the results of model creation. During the evaluation process, the results from the two feature extraction methods, FastText and TF-IDF, will be compared to determine which one performs better. The best model will then be used to analyze sentiment regarding the Kampus Merdeka program. The model evaluation will also include testing using a confusion matrix. A confusion matrix is a table that represents the classification of correct test data and incorrect test data. The confusion matrix is used to validate the results of the testing conducted. The depiction and explanation of the confusion matrix are presented in table 1.

Table 1. Confusion Matrix

	Predicted Positive	**Predicted Negative**
Actual Positive	TP	FP
Actual Negative	FN	TN

 ISSN 2723-6471

The parameters that will be used to validate the results of this research are three, namely precision, recall, and accuracy[20]. The Precision metric describes the level of accuracy between the actual data and the predicted results produced by the model. It measures the proportion of true positives (TP) out of the total number of instances that were predicted as positive, including both true positives (TP) and false positives (FP). The formula for precision is given in Equation (8).

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{8}$$

Next, the Recall metric indicates the effectiveness of the model in retrieving relevant information. It is calculated by taking the ratio of true positives (TP) to the sum of true positives (TP) and false negatives (FN), as shown in Equation (9).

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{9}$$

Lastly, Accuracy measures the overall correctness of the model by calculating the proportion of true positives (TP) and true negatives (TN) out of all predictions made, which include true positives, true negatives, false positives (FP), and false negatives (FN). The formula for accuracy is represented in Equation (10).

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{10}$$

2.6. The Sentiment Analysis Results

The sentiment analysis results are a stage where the best model obtained will be used to analyze sentiment in each of the Kampus Merdeka programs. In this research, the analysis is focused on three programs with the highest interest: the internship program, the independent study program, and the teaching program. For each program, tweets are collected and categorized into positive, neutral, or negative sentiments based on public opinion. The sentiment distribution is analyzed across different time periods to identify how public perception has evolved. The internship program generally shows a higher proportion of positive sentiment, indicating that students appreciate the practical experience offered. Meanwhile, the independent study and teaching programs also show a trend of positive sentiment, albeit with occasional concerns related to implementation and guidance. Overall, the analysis reveals that the Kampus Merdeka programs are well-received, providing useful insights for policymakers to refine and enhance future implementations based on public feedback.

3. Results and Discussion

The result of this research is a sentiment analysis model created using the best feature extraction algorithm from the comparison between TF-IDF and FastText. The following is the result of the discussion from the conducted research.

3.1. Data Collection

The data collection is divided into two parts. The first is used to gather data that will later be used to build the sentiment model. The second is used for sentiment analysis on the Kampus Merdeka program. The tweets were collected from the period of 2020-2023, with a total of 16,730 data points used for model building and 13,771 data points used for sentiment analysis testing. The data covers three Kampus Merdeka programs, with 4,007 data points for the internship program, 3,444 data points for the independent study program, and 6,318 data points for the teaching program. Table 2 shows the results of the data scraping used for building the sentiment analysis model.

Table 2. Scraping Data Results for Model Building

No	Date	Tweet
0	22 Maret 2020	PERBINCANGAN TARIKH PBAKL 2020. Dengan ini dimaklumkan bahawa MBKM telah memutuskan Tarikh penganjuran PBAKL 2020 ditetapkan pada 17 hingga 26 Julai 2020, di World Trade Centre KL. Urusetia Tetap Majlis Buku Kebangsaan Malaysia 17 Mac 2020 11:30 pagi
1	20 Maret 2020	@AnisaW05 gejek yaa mbkm mantep
…	....	....

 ISSN 2723-6471

16729	21 April 2023	Kampus harus menjadi laboratorium untuk mengkader pemimpin bangsa dan negara, sehingga mesti merdeka dan memiliki otonomi untuk memilih

Next, the data that will be used for sentiment analysis model prediction is presented, where this data covers three programs from the years 2020 to 2023. Table 3, table 4, and table 5 show the results of the data scraping.

Table 3. Scraping Results for Test Data of the Internship Program

No	Date	Tweet
0	30 Agustus 2021	gue mau ikut kampus merdeka, tapi bingung mau ambil magang atau student exchange
1	30 Agustus 2021	@ngen_uh ini magang yg dari kampus merdeka atau mandiri
…	....	....
4007	21 April 2023	ke insecurean liat org org pada magang mbkm

Table 4. Scraping Results for Test Data of the Independent Study Program

No	Date	Tweet
0	22 Agustus 2021	Astaga punya teman di studi independen :)
1	22 Agustus 2021	Zombie mode on 1. Kuliah dari pagi-sore (smt 5) 2. Studi independen 3. Organisasi 4. Kerja jam 5 sore sampe 12 malam.
…	....	....
3444	5 Maret 2023	@tikadwiyy Perkenalkan kami peserta Studi Independen (RevoU x Kampus Merdeka Batch FEB'23) dari Team 10 Section Bhinneka, saat ini sedang melakukan penelitian yang *bertujuan untuk mengetahui masalah-masalah yang user hadapi saat menggunakan aplikasi RedDoorz.* - continued

Table 5. Scraping Results for Campus Teaching Program Test Data

No	Date	Tweet
0	21 Juni 2021	Haii #dahlanmuda, berikut informasi syarat Program Kampus Mengajar #2 untuk dapat direkognisi ke nilai KKN ðŸ˜Š - Source: @kknuad.official - #universitasahmaddahlan #weareuad #muhammadiyah #dahlanmuda #lppmuad #kknkampusmengajar2 #kampusmengajar https://t.co/q4jSKRFXGG
1	21 Juni 2021	Mahasiswa Perlu Tahu, Benefit Mengikuti Program Kampus Mengajar Kemendikbud: Program Kampus Mengajar ditujukan bagi mahasiswa ini memiliki berbagai manfaat untuk mahasiswa itu sendiri. https://t.co/GltGmgmNnb https://t.co/UF6xAGehMe
…	....	....
6318	15 Juni 2023	bismillah bsk mau retake tes kampus mengajar. wish me luck!! semoga dilancarkan tanpa ada kendala lagi.

3.2. Labeling Data

The data used for model building, as shown in table 3, will be labelled in each row with positive, negative, and neutral labels. This process is done manually by reading each tweet one by one and assigning the appropriate label. Additionally, tweets that are not relevant to the Kampus Merdeka discussion or those that are meaningless will be removed. The labelling process resulted in a dataset of 3,445 entries. Table 6 provides an example of the data labelling results.

Table 6. Example of Data Labelling

Sentimen	Tweet
negatif	@lagiminumair Nadiem kan menjanjikan 'kampus merdeka' jadi katanya syaratnya dipermudah kan buat jadi ptn bh. tapi aku malah gatau kalo kampusku mau jadi ptn bh
positif	@tanyakanrl awal magang kampus merdeka, seru bggggttt.

 ISSN 2723-6471

netral	Ada yg namanya Kampus Merdeka sebagai pengganti skripsi, dan status program ini tidak wajib untuk tiap univ. Tanyakan pada kampusmu bagaimana program ini, apakah bisa berjalan, bagaimana mekanismenya.

3.3. Text Preprocessing

The text preprocessing steps are performed with the help of the NLTK library. The steps involved in text preprocessing are explained below.

Cleaning: This step involves removing special characters (@#$%^&*()_+!~|?) from tweets. In addition, usernames, hashtags, and URLs are also removed. The Example of Cleaning Result is shown in table 7.

Table 7. Example of Cleaning Result

Tweet	Cleaning
@lagiminumair Nadiem kan menjanjikan 'kampus merdeka' jadi katanya syaratnya dipermudah kan buat jadi ptn bh. tapi aku malah gatau kalo kampusku mau jadi ptn bh	Nadiem kan menjanjikan kampus merdeka jadi katanya syaratnya dipermudah kan buat jadi ptn bh tapi aku malah gatau kalo kampusku mau jadi ptn bh

Case Folding: In this step, uppercase letters are converted into lowercase letters. Below is an example of the case folding result in the model-building data shown in table 8.

Table 8. Example of Casefolding Result

Tweet	Casefolding
Nadiem kan menjanjikan kampus merdeka jadi katanya syaratnya dipermudah kan buat jadi ptn bh tapi aku malah gatau kalo kampusku mau jadi ptn bh	nadiem kan menjanjikan kampus merdeka jadi katanya syaratnya dipermudah kan buat jadi ptn bh tapi aku malah gatau kalo kampusku mau jadi ptn bh

Tokenizing: This is the process of breaking down text into smaller parts by removing numbers, spaces, punctuation marks, and other characters that are not used and do not affect data processing. The Example of Tokenizing Result is shown in table 9.

Table 9. Example of Tokenizing Result

Tweet	Tokenizing
nadiem kan menjanjikan kampus merdeka jadi katanya syaratnya dipermudah kan buat jadi ptn bh tapi aku malah gatau kalo kampusku mau jadi ptn bh	['nadiem', 'kan', 'menjanjikan', 'kampus', 'merdeka', 'jadi', 'katanya', 'syaratnya', 'dipermudah', 'kan', 'buat', 'jadi', 'ptn', 'bh', 'tapi', 'aku', 'malah', 'gatau', 'kalo', 'kampusku', 'mau', 'jadi', 'ptn', 'bh']

Normalization: In the word normalization process, a dictionary containing non-standard words is used. This dictionary is obtained from the Colloquial Indonesian Lexicon, consisting of 3,592 words, which was manually created. The Example of Normalization Result is shown in table 10.

Table 10. Example of Normalization Result

Tweet	Normalization
['nadiem', 'kan', 'menjanjikan', 'kampus', 'merdeka', 'jadi', 'katanya', 'syaratnya', 'dipermudah', 'kan', 'buat', 'jadi', 'ptn', 'bh', 'tapi', 'aku', 'malah', 'gatau', 'kalo', 'kampusku', 'mau', 'jadi', 'ptn', 'bh']	['nadiem', 'kan', 'menjanjikan', 'kampus', 'merdeka', 'jadi', 'katanya', 'syaratnya', 'dipermudah', 'kan', 'buat', 'jadi', 'ptn', 'bh', 'tapi', 'aku', 'malah', 'enggak', 'tau', 'kalo', 'kampusku', 'mau', 'jadi', 'ptn', 'bh']

Stopword Removal: Stopword removal is the process of eliminating words that are considered unimportant in the text. In this research, Indonesian stopwords from the Natural Language Toolkit (NLTK) were used, and some words that were deemed important in the stopword list, which determine negative sentiment, such as 'tidak' (not), 'bahkan' (even), and 'tapi' (but), were kept. The Example of Stopword removal results is shown in table 11.

 ISSN 2723-6471

Table 11. Example of Stopword Removal Result

Tweet	Stopword Removal
['nadiem', 'kan', 'menjanjikan', 'kampus', 'merdeka', 'jadi', 'katanya', 'syaratnya', 'dipermudah', 'kan', 'buat', 'jadi', 'ptn', 'bh', 'tapi', 'aku', 'malah', 'enggak tau', 'kalo', 'kampusku', 'mau', 'jadi', 'ptn', 'bh']	['nadiem', 'menjanjikan', 'kampus', 'merdeka', 'syaratnya', 'dipermudah', 'ptn', 'bh', 'tapi', 'gatau', 'kalo', 'kampusku', 'ptn', 'bh']

Stemming: In this research, the stemming process is carried out using the Sastrawi library, by converting affixed words into their root form. The example of stemming result is shown in table 12.

Table 12. Example of Stemming Result

Tweet	Stemming
['nadiem', 'kan', 'menjanjikan', 'kampus', 'merdeka', 'jadi', 'katanya', 'syaratnya', 'dipermudah', 'kan', 'buat', 'jadi', 'ptn', 'bh', 'tapi', 'aku', 'malah', 'enggak tau', 'kalo', 'kampusku', 'mau', 'jadi', 'ptn', 'bh']	['nadiem', 'janji', 'kampus', 'merdeka', 'syarat', 'mudah', 'ptn', 'bh', 'tapi', 'gatau', 'kalo', 'kampus', 'ptn', 'bh']

3.4. Feature Representation

After data preprocessing is completed, the next step is to perform feature representation by converting the data into numeric form so it can be input into the training model.

3.4.1. TF-IDF

In the TF-IDF process, data is transformed from words into numeric values. The result of this process will take the form of a matrix, with the data as rows and features as columns. This process is carried out using the scikit-learn library, with the generated word features amounting to 5,974. Table 13 shows the results of feature representation using TF-IDF.

Table 13. TF-IDF Result

Location	TF-IDF
(0, 2510)	**0.15**
(0, 5342)	0.18
(0, 1539)	0.10
(0, 5331)	0.12
....	....
(3445, 3419)	0.11
(3445, 2519)	0.16

3.4.2. Fasttext

In the training process of this research, the model was trained using the skip-gram architecture for 1,000 epochs, and the word embedding dimension used was 200 dimensions. After the model is trained, word embeddings are formed, representing the vector of a word. Words with similar meanings are expected to have vectors that are close to each other. Table 14 shows words that have similar meanings to the word "mahasiswa" (student).

Table 14. The list of similar word

Word	Words that have similarity
Mahasiswa	[('kampus', 0.56), ('merdeka', 0.55), ('mahasiwa', 0.53), ('ajar', 0.45), ('magang', 0.40)]

 ISSN 2723-6471

The next step is to convert the sentences/documents in the dataset into vectors that represent the sentences/documents. This is done by summing all the normalized word vectors that make up a document and averaging them. Figure 5 shows the visualization of how a document vector is formed from the vectors of its constituent words.

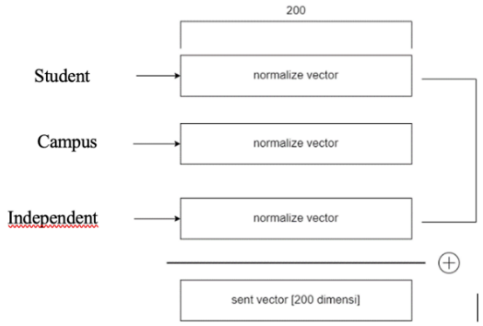

Figure 5. Visualization of Vector Formation

Table 15 shows the results obtained from the FastText calculation.

Table 15. Fasttext Result

Document -	Vector-			
	1	**2**	**….**	**200**
0	-0.001	0.009	….	-0.021
1	0.009	-0.030	….	-0.019
2	-0.043	-0.051	….	-0.031
….	….	….	….	….
3445	-0.001	-0.021	….	0.011

3.5. Model Develop

The data that has gone through the feature extraction process will then undergo classification using the SVM algorithm. In the model-building process, several steps and parameters are tested to evaluate how they enhance the model's accuracy. To achieve the best accuracy, there are several hyperparameter tunings that can be adjusted in SVM. Below are the hyperparameters used to determine the best parameter combination in the SVM model, as shown in table 16.

Table 16. Hyperparameter

Hyperparameter	Value
Kernel	RBF, Linear, Polynomial
C	0,001-1000
Gamma	0,001-1000

3.6. Model Evaluation

The next step in this research is to perform model evaluation. The model evaluation is carried out to determine how well the created model performs in classifying tweets. After the training process from the previous step, the model results were obtained. The model results in this study are divided into two, one using TF-IDF and the other using FastText. The results of the training model with TF-IDF can be seen in table 17.

Table 17. Training model results using TF-IDF

Dataset Ratio	Kernel	Accuracy	Precision	Recall
60:40	Linear	71%	76%	71%
70:30	Poly	70%	78%	70%
80:20	RBF	72%	81%	72%

From table 17, it can be seen that the best training model ratio using TF-IDF is achieved with the best value at an 80:20 dataset ratio, with an accuracy reaching 72%, and the best kernel using linear RBF. Next, in table 18, the training model using FastText is shown.

Table 18. Training model results using FastText

Dataset Ratio	Kernel	Accuracy	Precision	Recall
60:40	RBF	70%	77%	70%
70:30	RBF	72%	75%	71%
80:20	RBF	73%	81%	73%

From table 18, it can also be seen that the best ratio in the training model using FastText is 80:20, with the best kernel using RBF. The results from both training models show that FastText outperforms TF-IDF.

3.7. Sentiment Analysis Results

The developed model will then be used to predict sentiment in the Kampus Merdeka program. The prediction data used has gone through preprocessing, including stemming. Before making predictions, the Kampus Merdeka program data will first be divided based on the program's timeline from 2021 to 2023, to clearly understand the sentiment of each program during its respective period. The timeline of each program, as created based on information from the Kampus Merdeka website, can be seen in table 19 and table 20.

Table 19. Timeline for Internship and Independent Study

Batch Number-	Time
Batch 1	August 2021 - December 2021
Batch 2	February 2022 - June 2022
Bacth 3	August 2022 - December 2022
Batch 4	February 2023 - June 2023

Table 20. Timeline for Teaching Campus

Batch Number-	Time
Batch 1	March-June 2021
Batch 2	August-December 2021
Batch 3	January-June 2022
Batch 4	August - December 2022

After being divided according to the timeline, the data will next be converted into vector form. The transformation of data into vectors will be carried out following the structure of the FastText model that has been created. Afterwards, the data will be imported into the model which has previously been converted into a pickle format. The following are the prediction results that have been successfully conducted, as shown in table 21, table 22, and table 23.

Table 21. Kampus Merdeka Internship Program

Program Periods	Negative	Neutral	Positive
August -December 2021	473	45	790
February-June 2022	340	31	665
August-December 2022	264	20	549
February-June 2023	264	8	558

Table 22. Kampus Merdeka Teaching Program

Program Periods	Negative	Neutral	Positive
March-June 2021	260	41	693
August-December 2021	385	64	860
January-June 2022	418	60	892

 ISSN 2723-6471

August-December 2022	383	63	856

Table 23. Kampus Merdeka Independent Study Program

Program Periods	Negative	Neutral	Positive
August-December 2021	129	8	636
February-June 2022	155	5	1026
August-December 2022	126	3	591
February-June 2023	116	1	650

The Kampus Mengajar program showed predominantly positive sentiment, with 67% of tweets expressing positive opinions, 29% negative, and 4% neutral. Sentiment analysis from 2020 to 2023 revealed that overall, public sentiment toward the Kampus Merdeka program was positive, but fluctuated in response to policy changes. Positive sentiment peaked at 68% during the initial launch in early 2020, driven by support for new learning opportunities. However, by mid-2021, positive sentiment fell to 45%, while negative sentiment rose to 40% due to unclear guidelines and access barriers. By late 2022, positive sentiment recovered to 65% following policy adjustments that were well-received. Neutral sentiment remained stable at around 10%, mostly reflecting factual statements. These findings indicate that public sentiment is sensitive to policy changes, highlighting the need for ongoing monitoring. Future research should explore event-based sentiment analysis to better understand these fluctuations.

4. Discussion

The comparison in this study between FastText and TF-IDF for feature representation demonstrated that FastText slightly outperformed TF-IDF in terms of overall model performance. Specifically, FastText achieved an accuracy of 73% compared to 72% for TF-IDF. In addition to accuracy, other performance metrics were used to evaluate and compare the models, including precision, recall, and F1-score. For FastText, the model achieved a precision of 81%, a recall of 72%, and an F1-score of 76%. In contrast, the TF-IDF model resulted in a precision of 81%, a recall of 71%, and an F1-score of 75%. The slightly higher F1-score achieved by FastText suggests that it is better suited for handling the imbalance between positive, neutral, and negative sentiment classes, which is often encountered in social media data. This difference can be attributed to FastText's ability to capture semantic relationships between words through subword embeddings, leading to better contextual representation. Meanwhile, TF-IDF, although effective in representing term frequency, is limited in handling word variations and semantic context. The sentiment analysis results show the distribution of tweets across three sentiment categories: positive, negative, and neutral. Out of the total tweets analyzed, 13,771 data points were used to evaluate the sentiment distribution for the Kampus Merdeka program. Table 24 below provides a detailed breakdown of the number of tweets in each sentiment category.

Table 24. The detailed breakdown of the number of tweets in each sentiment category

Sentiment Category	Number of Tweets	Percentage
Positive	8,357	60.7%
Negative	4,608	33.5%
Neutral	806	5.8%

This breakdown indicates that the overall sentiment toward the Kampus Merdeka program is predominantly positive, with the majority of tweets expressing support or appreciation for the program's impact. Negative sentiment, though significantly lower, still represents a considerable portion, highlighting some areas of public dissatisfaction or criticism. Neutral tweets, which account for only 5.8%, consist of factual statements or observations that do not express a clear opinion toward the program. The detailed analysis of these categories is crucial for understanding the sentiment landscape and provides insights into how the public perceives different aspects of the Kampus Merdeka program. For instance, the high proportion of positive tweets can be attributed to the perceived benefits of the program, such as opportunities for internships and independent studies, as frequently mentioned in the positive tweets. Meanwhile, the negative tweets often focused on implementation issues or concerns about the program's accessibility, indicating areas for potential improvement. This breakdown also highlights the importance of accurate classification, particularly for

ISSN 2723-6471

neutral tweets. Although the number of neutral tweets is relatively small, incorrect classification of these tweets as either positive or negative could skew the overall sentiment analysis. Therefore, special attention was given during the manual labeling process to ensure that only tweets with a clear lack of sentiment were categorized as neutral. With this distribution in mind, it is evident that the overall public perception of the Kampus Merdeka program is favorable. However, understanding the specific areas where negative sentiment arises can provide valuable feedback for policymakers to further refine and enhance the program.

The parameters tested, including the values of C, gamma, and the choice of kernel in SVM, contributed to optimizing the model's performance. This finding is consistent with previous research, such as the study by [6], which compared FastText and Word2Vec, showing that FastText achieved a higher accuracy of 93% compared to Word2Vec's 92%. This supports the notion that FastText is more effective in understanding the semantic and contextual relationships between words, especially when dealing with rare or unseen words. FastText's ability to leverage character-level information through n-gram representations is particularly advantageous in handling the variability and complexity of natural language, which is common in social media data. The advantage of FastText, as demonstrated in both this and prior studies, lies in its capability to capture deeper and broader word associations, allowing it to model context more effectively than traditional methods like TF-IDF. This makes it a powerful tool for text classification tasks, including sentiment analysis, where understanding subtle word meanings and relationships is crucial for accurate prediction. By leveraging its subword-based approach, FastText provides a robust solution for overcoming the limitations of word sparsity and unseen vocabulary that often challenge traditional bag-of-words models.

Processing Indonesian tweets poses unique challenges for sentiment analysis due to the informal and dynamic nature of the language used on social media and the frequent mixing of Indonesian and English (code-switching). This complexity affects text preprocessing and feature extraction. Colloquial expressions, abbreviations, and slang such as "btw kampus ini keren bgt" or "nggak paham deh dgn program ini" often deviate from standard Indonesian, leading to misinterpretations. Code-switching, where Indonesian and English phrases are mixed in a single tweet, further complicates the feature extraction process, as certain terms may be misclassified. Additionally, users frequently use shortened words like "bgt" (for "banget") or create new phonetic variants such as "ampe" (for "sampai"), which makes normalization difficult due to the ever-evolving nature of online slang. These linguistic challenges increase the likelihood of misclassification, particularly for neutral or ambiguous tweets. While using FastText's subword embeddings helps capture these variations better than TF-IDF, some informal or mixed-language phrases still pose difficulties, indicating a need for more advanced context-aware models like BERT. This research highlights these challenges and suggests areas for future refinement, such as using context-sensitive embeddings and expanding normalization techniques to include new slang terms.

5. Conclusion

The sentiment analysis model developed in this research was successfully applied to predict public sentiment toward the three main Kampus Merdeka programs: internships (magang), independent studies (studi independen), and the Kampus Mengajar program. The model development process involved selecting the most appropriate feature extraction method by comparing two popular techniques: TF-IDF and FastText. Based on the analysis results, FastText outperformed TF-IDF, achieving an accuracy of 73%, precision of 81%, recall of 72%, and an F1-score of 76%. In comparison, TF-IDF achieved an accuracy of 72%, with a precision of 81% and a recall of 72%. The advantage of FastText can be attributed to its ability to capture deeper contextual relationships between words, enabling the model to better understand the semantic connections between terms. Public sentiment toward the Kampus Merdeka program was found to be predominantly positive. This is evidenced by the dominance of positive sentiment toward the three programs during the period from 2020 to 2023. These findings indicate that the implementation of the Kampus Merdeka program has been well-received by the public, particularly by students, and has had a positive impact on higher education in Indonesia.

6. Declarations

6.1. Author Contributions

Conceptualization: L.A., N.H., N., and M.F.A.; Methodology: N.H.; Software: L.A.; Validation: L.A. and N.H.; Formal Analysis: L.A. and N.H.; Investigation: L.A.; Resources: N.H.; Data Curation: N.H., N., and M.F.A.; Writing Original Draft Preparation: L.A., N.H., N., and M.F.A.; Writing Review and Editing: N.H. and L.A.; Visualization: L.A.; All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] I. Deffinika, I. W. Putri, and K. B. Angin, "Higher Education and Training Towards Global Competitiveness and Human Development in Indonesia," *Geojournal of Tourism and Geosite*, vol. 38, no. 4, pp. 1280–1288, 2021, doi: 10.30892/gtg.38436-770.

[2] H. Suharman and N. Hidayah, "Essentials of Intellectual Capital to Create Higher Education Sustainable Competitive Advantage: Environment Uncertainty as a Moderating Variable in Indonesia Private Universities," *International Journal of Economics and Business Administration*, vol. IX, no. 1, pp. 382–391, 2021, doi: 10.35808/ijeba/680.

[3] R. W. Threnisa, "Evaluation of Internship Program Kampus Merdeka in the Community-Based Total Sanitation Facilitator Field at Community Health Center Dr. Soetomo," *World Journal of Advanced Research and Reviews*, vol. 22, no. 2, pp. 1383–1386, May 2024, doi: 10.30574/wjarr.2024.22.2.1524.

[4] F. Fauzi, S. B. Abdinagoro, R. Kartono, A. Furinto, and M. Hamsal, "Extracting Public Opinion and Popularity of Islamic Bank in Indonesia: A Big Data of Social Media and Google Trends Approach," *E3S Web of Conferences*, vol. 426, no. 02019, pp. 1-7, 2023, doi: 10.1051/e3sconf/202342602019.

[5] M. M. Zaheer and P. Nirmala, "An Efficient Approach to Detect Liver Disorder Using Customised SVM in Comparison with Random Forest Algorithm to Measure Accuracy," *Cardiometry*, Vol. 25, no 12, pp. 1024–1030, Feb. 2023, doi: 10.18137/cardiometry.2022.25.10241030.

[6] A. Zakaria, R. R. Isnanto, and O. D. Nurhayati, "Particle Swarm Optimization and Support Vector Machine for Vehicle Type Classification in Video Stream," *Int. J. Comput. Appl.*, vol. 182, no. 18, pp. 9–13, Sep. 2018, doi: 10.5120/ijca2018917880.

[7] M. M. Truşcă, "Efficiency of SVM Classifier with Word2Vec and Doc2Vec Models," in *Proc. Int. Conf. Appl. Stat.*, vol. 1, no. 1, pp. 496–503, Oct. 2019, doi: 10.2478/icas-2019-0043.

[8] A. K. Tegegnie, A. N. Tarekegn, and T. A. Alemu, "A Comparative Study of Flat and Hierarchical Classification for Amharic News Text Using SVM," *Int. J. Inf. Eng. Electron. Bus.*, vol. 9, no. 3, pp. 36–42, May 2017, doi: 10.5815/ijieeb.2017.03.05.

[9] I. Sharif and D. Chaudhuri, "A multiseed-based SVM classification technique for training sample reduction," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 27, no. 1, pp. 595–604, 2019, doi: 10.3906/elk-1801-157.

[10] Z. H. Kilimci and R. Duvar, "An Efficient Word Embedding and Deep Learning Based Model to Forecast the Direction of Stock Exchange Market Using Twitter and Financial News Sites: A Case of Istanbul Stock Exchange (BIST 100)," *IEEE Access*, vol. 8, no 10, pp. 188186–188198, 2020, doi: 10.1109/ACCESS.2020.3029860.

 ISSN 2723-6471

[11] J. Mutinda, W. Mwangi, and G. Okeyo, "Sentiment Analysis of Text Reviews Using Lexicon-Enhanced Bert Embedding (LeBERT) Model with Convolutional Neural Network," *Applied Sciences (Switzerland)*, vol. 13, no. 3, pp. 1-14, Feb. 2023, doi: 10.3390/app13031445.

[12] H. Wei, G. Lin, L. Li, and H. Jia, "A context-aware neural embedding for function-level vulnerability detection," *Algorithms*, vol. 14, no. 11, pp. 1-20, Nov. 2021, doi: 10.3390/a14110335.

[13] Z. H. Kilimci and R. Duvar, "An efficient word embedding and deep learning based model to forecast the direction of stock exchange market using twitter and financial news sites: A case of istanbul stock exchange (BIST 100)," *IEEE Access*, vol. 8, no. 10, pp. 188186–188198, 2020, doi: 10.1109/ACCESS.2020.3029860.

[14] T. Dodiya, "Using Term Frequency - Inverse Document Frequency to find the Relevance of Words in Gujarati Language," *Int J Res Appl Sci Eng Technol*, vol. 9, no. 4, pp. 378–381, Apr. 2021, doi: 10.22214/ijraset.2021.33625.

[15] Z. Jiang, B. Gao, Y. He, Y. Han, P. Doyle, and Q. Zhu, "Text Classification Using Novel Term Weighting Scheme-Based Improved TF-IDF for Internet Media Reports," *Math Probl Eng*, vol. 2021, no. 3, pp. 1-30, 2021, doi: 10.1155/2021/6619088.

[16] V. Vichianchai and S. Kasemvilas, "A New Term Frequency with Gaussian Technique for Text Classification and Sentiment Analysis," *Journal of ICT Research and Applications*, vol. 15, no. 2, pp. 152–168, Oct. 2021, doi: 10.5614/itbj.ict.res.appl.2021.15.2.4.

[17] D. Meidelfi, I. Rahmayuni, T. Hidayat, and D. Chandra, "TF-IDF Implementation for Similarity Checker on The Final Project Title," *International Journal of Advanced Science Computing and Engineering*, vol. 3, no. 1, pp. 40–52, 2021, doi : https://doi.org/10.62527/ijasce.3.1.3.

[18] F. Q. Pei, D. B. Li, Y. F. Tong, and F. He, "Process Service Quality Evaluation Based on Dempster-Shafer Theory and Support Vector Machine," *PLoS One*, vol. 12, no. 12, pp. 1-16, Dec. 2017, doi: 10.1371/journal.pone.0189189.

[19] Z. Qian, Y. Gu, and W. Hong, "An Image Tampering Detection Algorithm of Qualification Certificate Based on CNN and SVM," *Academic Journal of Computing and Information Science*, vol. 4, no. 7, pp. 24-38, 2021, doi: 10.25236/ajcis.2021.040705.

[20] B. Juba and H. S. Le, "Precision-Recall versus Accuracy and the Role of Large Data Sets," *Association for the Advancement of Artificial Intelligence*, vol. 33, no. 01, pp. 4039-4048, 2019, doi: https://doi.org/10.1609/aaai.v33i01.33014039.