Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control
Journal homepage: http://kinetik.umm.ac.id
ISSN: 2503-2267
Vol. 10, No. 3, August, Pp. 355-366

Classification of Livin' by Mandiri customer satisfaction using MLP with BM25 and TF-IDF feature weighting

Aina Mardiah[1], Salsa Dillah[1], Dewi Fatmarani Surianto*[1], Nur Fadilah[2], Satria Gunawan Zain[1]
Makassar State University, Makassar[1]
Megarezky University Makassar, Makassar[2]

Article Info
Keywords:
Customer Satisfaction, Multi-Layer Perceptron, BM25, TF-IDF, Livin' Mandiri

Article history:
Received: February 05, 2025
Accepted: June 02, 2025
Published: August 31, 2025

Cite:
A. Mardiah, S. Dillah, D. F. Surianto, N. Fadilah, and S. G. Zain, "Classification of Livin' by Mandiri Customer Satisfaction Using MLP with BM25 and TF-IDF Feature Weighting", *KINETIK*, vol. 10, no. 3, Aug. 2025.
https://doi.org/10.22219/kinetik.v10i3.2248

*Corresponding author.
Dewi Fatmarani Surianto
E-mail address:
dewifatmaranis@unm.ac.id

Abstract
The increasing use of mobile banking applications such as Livin' by Mandiri requires an analysis of customer satisfaction based on user reviews. This study classifies customer satisfaction levels using the Multi-Layer Perceptron (MLP) algorithm with two feature extraction methods, namely BM25 and TF-IDF. A total of 1,143 reviews were collected from the Google Play Store and App Store. Three test scenarios were applied: (1) comparison of feature extraction methods, (2) application of Synthetic Minority Over-Sampling Technique (SMOTE), and (3) application of Synonym Replacement-based Easy Data Augmentation (EDA) technique. The evaluation results show that the combination of BM25 and data augmentation produces the highest performance, with 97% accuracy and 98% precision, recall, and F1-score, respectively. BM25 proved to be more effective in understanding the context of reviews, while data augmentation improved the quality of representation, especially for minority classes such as neutral sentiment. These findings make a significant contribution to the improvement of Livin' by Mandiri digital services and serve as a reference for the development of review-based satisfaction classification systems in the digital banking sector.

1. Introduction

In the digital era, the internet plays an important role in the daily lives of most Indonesians. It serves as a medium for sharing information without spatial and temporal restrictions, accessible anywhere and anytime as long as there is an internet connection [1][2]. According to BPS data from 2022, 66.48% of the Indonesian population has used the internet, an increase from 62.10% in 2021. This condition impacts many aspects of life, including the banking industry. Banking companies continue to innovate to provide better and more efficient services to their customers due to advances in banking technology [3]. One such innovation is Livin' by Mandiri, a mobile banking application developed by PT Bank Mandiri (Persero) Tbk.

The Livin' by Mandiri application is a mobile banking platform from Bank Mandiri designed to provide a more personalized banking experience for customers. Through this application, customers can access various financial services, including credit cards, loan accounts, deposits, savings, and other services [4]. According to PT Bank Mandiri (Persero) Tbk, the number of Livin' Mandiri application users is expected to increase by 37% by May 2024. The number of user reviews also rises alongside with the number of users.

Reviews reflect varying levels of customer satisfaction, ranging from positive feedback, complaints about technical issues, application performance, and the quality of services provided [4]. Through these reviews, application developers can understand the strength and weaknesses of the application from the user's perspective, enabling improvements to be made to the application services [5][6]. In addition, reviews can be utilized by prospective application users to learn about the experiences of previous users [7].

In recent years, customer sentiment analysis has gained prominence in the field of digital banking, particularly as mobile banking applications become integral to financial transactions. Various machine learning approaches, especially those involving Natural Language Processing (NLP), have been employed to analyze customer feedback and improve user experience. State-of-the-art research increasingly combines classical machine learning with feature engineering techniques to enhance classification accuracy in sentiment prediction.

Research on the classification of user satisfaction levels based on reviews is increasingly relevant and has developed various approaches. Previous research on Livin' by Mandiri review data sourced from Kaggle, using Naïve Bayes resulted in an accuracy of 0.9737, a precision of 0.9327, and a recall of 0.9685 on training data, as well as an accuracy of 0.9737 on test data [5]. Another study on 1,637 tweets about Livin' by Mandiri, using TF-IDF and Multinomial Naïve Bayes, achieved 93% accuracy, 90% precision, 93% recall, and 91% F1-Score, showing the majority of positive sentiments [8]. An analysis of 4,719 Livin' by Mandiri user reviews on the Play Store with e-servqual-based Naïve Bayes showed 88.07% accuracy, with a dominant negative sentiment (64%) and the highest reliability dimension (47.47%) [9].

Research in 2024 on the Blu BCA application using Naïve Bayes resulted in an accuracy of 85.31%, with the majority of positive reviews despite complaints related to application performance and data security [10]. Furthermore, research on the Dana application, using Naïve Bayes and SVM, showed that SVM was superior with an accuracy of 92.78% and 84.7% of reviews reflecting negative sentiments [11]. Sentiment analysis research on hotel reviews with Naïve Bayes achieved 96.08% accuracy, detecting sentiment patterns useful for business decisions, and achieved 85.31% accuracy [12]. Further research compared SVM, Naïve Bayes, and Logistic Regression on 24,401 tweets about BSI Bank, where SVM achieved 88% accuracy, revealing 70% negative sentiment [13]. Research analyzing 228,208 tweets about the COVID-19 vaccine with Multi-Layer Perceptron (MLP) achieved 81.2% accuracy, with a sentiment distribution of 35% positive, 16.3% negative, and 48.7% neutral [14].

From previous research, it can be seen that various classification algorithms, such as Naïve Bayes, SVM, Logistic Regression, Multi-Layer Perceptron and K-Nearest Neighbor, are widely used for sentiment analysis and user satisfaction classification. However, there are still gaps that are the focus of this research. First, most studies rely on only one feature extraction or classification technique, whereas a more complex combination of methods is possible to improve classification performance [10]. Secondly, the selection of feature extraction techniques is an important element in the text classification process, as it directly affects classification performance and results [15]. Thirdly, previous studies tend to focus on one particular data source, such as the Google Play Store, so the limited context and perceptions of users outside the platform are not thoroughly represented. Therefore, this research will collect data from two different sources to overcome these issues.

A major challenge in user satisfaction classification is that conventional classification approaches tend to categorize sentiments only in general terms. That is, there have not been many systematic efforts to analyze user reviews in classifying overall user satisfaction levels [9]. Thus, this research aims to analyze and improve classification accuracy by developing a more systematic and comprehensive classification approach for user satisfaction levels based on reviews.

An approach that can be used to process and categorize text data is to combine text mining techniques and Artificial Neural Network (ANN) algorithms, such as Multi-Layer Perceptron (MLP), along with BM25 and TF-IDF weighting methods. MLP is effective for recognizing complex patterns that cannot be captured by linear models, thanks to its deep network structure and backpropagation learning process [16]. Meanwhile, BM25 weights words by considering Term Frequency (TF), Inverse Document Frequency (IDF), and document length, while TF-IDF assesses the significance of words based on their frequency in the document compared to the overall corpus frequency [17][18].

It is hoped that this research will support strategic decisions in developing Livin' by Mandiri's digital services, ultimately enhancing application quality, customer loyalty, and Bank Mandiri's reputation. However, prior studies often rely on single feature extraction or classification models, limiting contextual accuracy, especially in handling imbalanced and underrepresented sentiment classes. To address this, this study proposed a hybrid model combining MLP with BM25 and TF-IDF, further strengthened by SMOTE and synonym-based augmentation. This integrated approach offers a more robust and context-aware classification of customer satisfaction in real-world scenarios.

2. Research Method

In conducting research on the classification of Livin' by Mandiri user reviews, there are systematic stages that must be completed to finalize the research. The process can be illustrated through a flowchart that explains each step of the research, as shown in Figure 1 below.

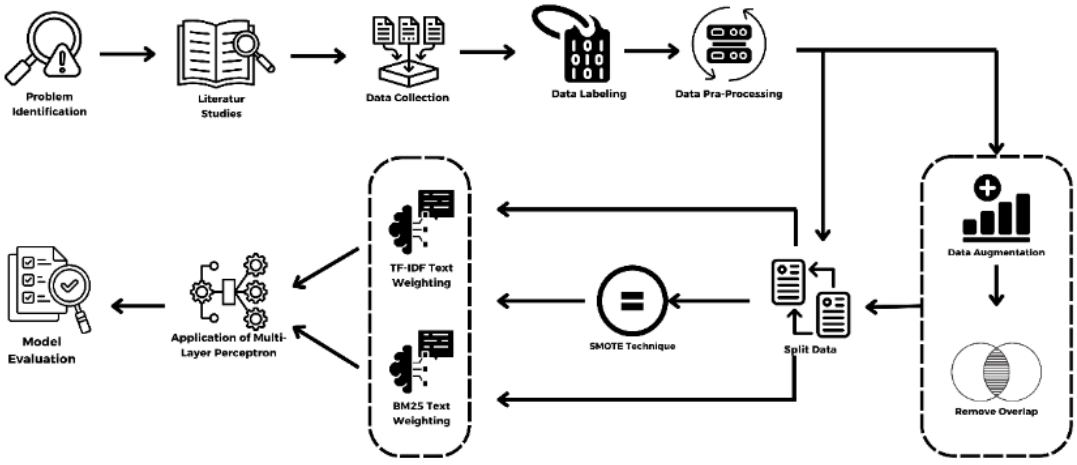


Figure 1. Research Stages

2.1 Problem Identification

This stage aims to gain an understanding of the challenges in classifying user satisfaction based on reviews. Observations were made on user reviews from the Play Store and App Store platforms to identify patterns that emerge in expressing customer satisfaction. One of the problems identified was the lack of a systematic and specific classification approach to comprehensively categorize user satisfaction levels, given that conventional approaches tend to only divide sentiments in general terms.

2.2 Data Collection

The data collection process involves gathering application reviews from the Google Play Store and Apple App Store. The results of the data collection are stored in Excel file format (.xlsx). The collection process was carried out from October 10, 2024, to October 24, 2024, with a total of 1,143 reviews collected. Of the 1,143 reviews, 671 reviews received negative ratings, 68 reviews received neutral ratings, and 405 reviews received positive ratings. An example of user reviews based on ratings can be seen in Table 1 below.

Table 1. User Reviews by Rating

No.	Reviews	Rating
1	*aplikasinya gak ribet, mudah, praktis untuk di pakai di hp ram kecil pun gak lemot sama sekali. Terima kasih livin sudah membantu saya dalam bertransaksi digital.*	5
2	*So far so good. Hanya saja appnya minta di update. Tetapi selalu tidak bisa terupdate, padahal saya menggunakan wifi*	3
3	*Please lah ini UI livin mau keliatan keren malah ribet dan pusing*	1

2.3 Data Labeling

In this stage, the sentiment class labeling process is performed for each review before proceeding to the word weighting and data sharing stages. The labeling is done automatically based on the rating given by the user. This labeling system uses three categories: 0 for negative, 1 for neutral, and 2 for positive. Reviews with ratings of 4 and 5 are automatically labeled as positive, while reviews with ratings of 1 and 2 are labeled as negative. For reviews that have a rating of 3, the labeling is done manually to determine whether the review is more likely to fall into the negative or positive class based on its text content. This labeling process covers a total of 1,143 classified reviews. An example of the results of labeling the review data is presented in Table 2 below.

Table 2. Review Data on Each Label

No.	Reviews	Label
1	*aplikasinya gak ribet, mudah, praktis untuk di pakai di hp ram kecil pun gak lemot sama sekali. terima kasih livin sudah membantu saya dalam bertransaksi digital.*	2
2	*So far so good. Hanya saja appnya minta di update. Tetapi selalu tidak bisa terupdate, padahal saya menggunakan wifi*	1
3	*Please lah ini UI livin mau keliatan keren malah ribet dan pusing*	0

2.4 Data Pre-Processing

The preparation of data before it is used in modeling is known as data pre-processing. This process aims to make the raw data format easier to understand [19] . Data pre-processing involves several steps to process the text. First, case folding is performed to convert all letters to lowercase. Then, stopword removal eliminates common words that are not important. Stemming converts words to their base form, and tokenizing breaks the text into small parts or tokens [7][20] . An example of a review that went through the text pre-processing stage is presented in Table 3.

Table 3. Text Pre-processing on Review Data

Stage	Reviews
Case folding	*aplikasinya gak ribet, mudah, praktis untuk di pakai di hp ram kecil pun gak lemot sama sekali. terima kasih livin sudah membantu saya dalam bertransaksi digital*
Stopword removal	*aplikasinya gak ribet, mudah, praktis pakai hp, ram gak lemot sekali. terima kasih livin membantu bertransaksi digital.*
Stemming	*aplikasi gak ribet, mudah, praktis pakai hp, ram gak lot sekali, terima kasih livin bantu transaksi digital.*

Stage	Reviews
Tokenizing	*['aplikasi', 'gak', 'ribet', 'mudah', 'praktis', 'pakai', 'hp', 'ram', 'gak', 'lot', 'sekali', 'terima', 'kasih', 'livin', 'bantu', 'transaksi', 'digital']*

2.5 Data Splitting

Data splitting is the process of dividing a data set into two parts. The training data is used to train the machine learning model, while the test data is used to evaluate the performance of the machine learning model [21]. In this research, the data splitting process uses a proportion of 80% training data and 20% test data, resulting in 928 training data points and 233 test data points from a total dataset of 1,161 data points. In addition, the use of random_state 42 aims to ensure the same division results every time the code is run, even if it is done repeatedly [22][23].

2.6 Imbalanced Data Technique (SMOTE)

The Synthetic Minority Over-Sampling Technique (SMOTE) adds samples to minority classes to produce balanced data [21]. The SMOTE process starts with random data from the minority class and identifies their K nearest neighbors. Next, interpolation is used to collect new samples from data in neighboring regions. The initial data is added by multiplying the feature vector difference between it and its neighboring data by a random number ranging from 0 to 1. The result is synthetic data that lies along the line between the two data points [22].

2.7 Easy Data Augmentation (EDA)

Data augmentation is one of the techniques in text classification used to reduce overfitting by improving the performance of small datasets [24]. Easy Data Augmentation (EDA) is an augmentation technique that utilizes a paraphrasing approach, which consists of four techniques: Synonym Replacement, Random Deletion Random Insertion, and Random Swap [25]. This research uses the Synonym Replacement technique which involves replacing one or more words in a sentence with their synonyms. The length of the sentence and the constant (augmentation level) determine the number of words replaced [26].

2.8 Overlap Data Removal

Overlap data removal is the process of removing data that has similarities between training and test data to prevent data leakage. This step is performed after the data augmentation stage to ensure variation and independence between subsets. By applying this technique, it is expected that no training data overlaps with test data, ensuring that model evaluation can be maintained.

2.9 Feature Extraction with TF-IDF and BM25

This research uses two feature weighting techniques, namely TF-IDF and BM25, to compare the results and obtain a more accurate analysis. The following is the explanation:

2.9.1 TF-IDF

TF-IDF is a calculation that determines how important each word or term is in a document and in the entire corpus of documents. The goal is to give weight to more relevant words in a particular document based on how often the word appears in that document [12]. The formula for calculating TF-IDF is presented in Equation 1.

$$TF - IDF = TF \times IDF = TF \times log\left(\frac{n}{df}\right) \quad (1)$$

Description:

- TF (Term Frequency): The frequency of words that appear in a particular document.
- IDF (Inverse Document Frequency): How rarely the word appears in each document in the corpus (the rarer the higher).
- n: Number of documents in the corpus.

2.9.2 BM25

The BM25 method in search systems ranks document matching results based on their relevance to the searched queries. This method is recognized as one of the most superior in the "best match" category because it has an effective formula for providing the right ranking, presented in Equation 2. BM25 calculates three important factors in ranking the documents, one of which is identifying the length of the document, which is not accounted for by TF-IDF feature weighting [17].

$$BM25(d_j, q_{1:N}) = \sum_{i=1}^{N} IDF(q1) \frac{TF(qi, dj).(k+1)}{TF(q1, dj) + k.\left(1 + b + b.\frac{|dj|}{L}\right)} \quad (2)$$

With L described in the Equation 3, that is:

$$L = \frac{\sum i\,|di|}{N} \quad (3)$$

IDF is calculated using Equation 4 below:

$$IDF(qi) = \log \frac{N - DF(qi) + 0{,}5}{DF(qi) + 0{,}5} \quad (4)$$

Description:
- Qi: Keyword or term
- dj: Document
- $q_{1\ N}$: Query consisting of N or keywords (*qi*) searched for
- |dj|: Document length
- TF (q_i, d_j): Number of documents containing keyword *qi* in document *dj*
- IDF (q_i): Measures how important term *qi* is in the entire document collection.
- L: Document length range
- k: Constant used in the evaluation
- b: Effect of document length

2.10 Application of Multi-Layer Perceptron (MLP) Model

Multilayer Perceptron (MLP) is one of the most widely used types of artificial neural networks. MLP consists of several layers of nodes, which include input layer, hidden layer, and output layer [27]. The Multi-Layer Perceptron model was used in this study due to the small amount of data and light computational requirements [10]. In this research, the multi-layer perceptron algorithm is used to classify sentiment by utilizing various parameters, as shown in Table 4 below.

Table 4. MLP Parameter Configuration Used

Parameter	Configuration
Hidden Layer	128, 64
Activation Function	ReLU
Solver	adam
Learning Rate	0,001
Maximum Iteration	1000

The formulas of the multi-layer perceptron model used in this research are presented in Equations 5 and 6.

$$Y = f(sum_i^n = 1\ X_i W_i + b) \quad (5)$$

Description:
- Y: output of the MLP
- X: input from MLP
- F: activation function used for each ReLU neuron

$$f(x) = \max(0, x \quad (6)$$

- W: weights connecting each neuron
- b: bias

2.11 Model Evaluation

At this stage, the trained model is then used to make predictions on the test data. The prediction results are then evaluated using a confusion matrix, which calculates four main categories: True Positive (TP), True Negative (TN),

False Positive (FP), and False Negative (FN). Furthermore, the confusion matrix serves as a tool to calculate important metrics such as accuracy, precision, recall, and F1-Score.

3. Results and Discussion

In this study, classification model testing was carried out using the Multi-Layer Perceptron (MLP) model to classify the user review text of the Livin' by Mandiri application. This classification process used three test scenarios: the scenario of different feature weighting methods, the scenario of using the SMOTE technique, and the scenario of using the data augmentation technique.

3.1 Feature Extraction Scenario

In the first scenario, the review classification model was tested based on two feature weighting methods: TF-IDF and BM25. Table 5 below shows the model testing results.

Table 5. Model Testing Results on Different Feature Extraction Scenarios

No.	Feature Extraction Method	Class	Recall	Precision	F1-Score	Accuracy
1	TF-IDF	Negative	83%	74%	78%	70%
		Neutral	14%	33%	20%	
		Positive	55%	63%	59%	
2	**BM25**	**Negative**	**86%**	**79%**	**82%**	**74%**
		Neutral	**14%**	**20%**	**17%**	
		Positive	**64%**	**71%**	**67%**	

The TF-IDF feature extraction method yielded an overall accuracy of 70%, with the best performance on the negative class, fair on the positive class, and the lowest on the neutral class. Meanwhile, the BM25 feature extraction method recorded a higher overall accuracy of 74%. The order of performance between classes in BM25 is similar to TF-IDF but shows an increase in percentage in each class.

To conclude, in this scenario, BM25 outperforms TF-IDF in terms of accuracy and performance in both negative and positive classes. This advantage occurred due to BM25's ability to adjust word weights in a more balanced manner, considering document length and word occurrence frequency using a logarithmic saturation function. This approach prevents the overweighting of frequently occurring words, making it more effective, especially for short documents and unbalanced data. In contrast, TF-IDF relies solely on word frequency without taking into account both document length and weight saturation, which may lead to a decrease in accuracy, especially in long documents and when certain words are dominant [17][18]. The improvements in BM25 demontrate its robustness in managing short and informal reviews typical of app feedback. These findings reinforce its suitability in practical digital service evaluation contexts.

3.2 SMOTE Technique Usage Scenario

In this scenario, it is similar to the previous scenario, specifically regarding the difference in feature weighting, but with the addition of the imbalanced data technique (SMOTE). The SMOTE technique aims to balance the data by adding data to the minority class, ensuring that all classes have a balanced amount of data. Figure 2 shows the comparison of the data amount before and after applying the SMOTE method.

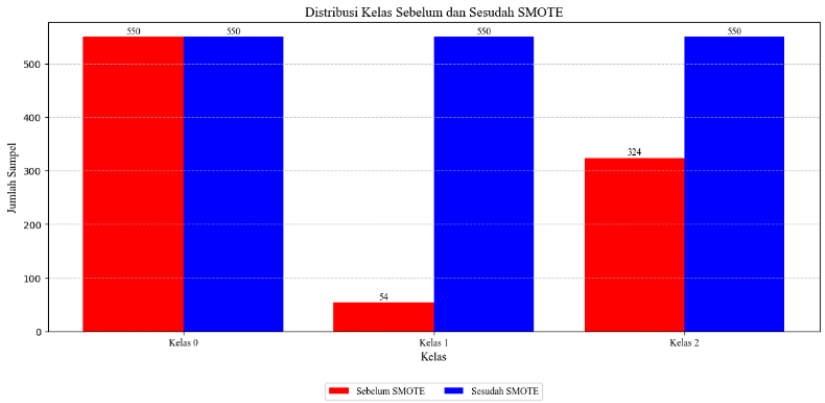


Figure 2. Class Distribution Before and After SMOTE

The SMOTE technique is applied after the data-splitting stage, meaning that only the training data, totaling 914 data, undergoes the SMOTE process. After applying SMOTE, the amount of data in each class was successfully balanced to 550 data resulting in a total of 1,143 data after SMOTE. Table 6 below presents the results of model testing for both feature extraction methods using the SMOTE technique.

Table 6. Testing Results of Feature Extraction Model with SMOTE Technique

No.	Feature Extraction Method	Technique	Class	Recall	Precision	F1-Score	Accuracy
1	TF-IDF	SMOTE	Negative	71%	77%	74%	65%
			Neutral	21%	12%	16%	
			Positive	62%	61%	62%	
		NON-SMOTE	**Negative**	**83%**	**74%**	**78%**	**70%**
			Neutral	**14%**	**33%**	**20%**	
			Positive	**55%**	**63%**	**59%**	
2	BM25	SMOTE	Negative	83%	77%	80%	70%
			Neutral	14%	12%	13%	
			Positive	59%	70%	64%	
		NON-SMOTE	**Negative**	**86%**	**79%**	**82%**	**74%**
			Neutral	**14%**	**20%**	**17%**	
			Positive	**64%**	**71%**	**67%**	

Based on Table 6, the TF-IDF method with SMOTE achieves an overall accuracy of 65%, while the BM25 method with SMOTE reaches an overall accuracy of 70%. The results indicate that performance on the neutral class has decreased. Compared to the first scenario, the use of SMOTE slightly improved balance in some classes, but not significantly for the neutral class. This suggests that while SMOTE helps with imbalanced data, it is not sufficiently effective for underrepresented classes.

SMOTE generates synthetic samples through interpolation between existing minority data; however, this approach tends to produce uniform samples that do not reflect the natural diversity of the minority classes. Consequently, the synthetic data generated carries a low risk of broadening the feature distribution, which can exacerbate the overfitting problem. Furthermore, if the available minority data is very limited and does not adequately represent the complexity of the class, the interpolation may become meaningless or even produce irrelevant noise, ultimately degrading the model performance [26]. Although SMOTE improved class balance, particularly for neutral reviews, its effect was limited due to the intrinsic semantic simplicity of synthetic samples. The failure to reflect actual linguistic variation restricted its utility, making SMOTE insufficient as a standalone solution for handling minority class.

3.3 Scenario of Data Augmentation Technique Usage

In this scenario, data augmentation techniques are applied after the data-splitting stage and are used for both feature extraction methods. The data augmentation technique employed is the Synonym Replacement technique, which replaces four words in a sentence with their synonyms to provide more inter-sentence variation and produce one new sentence variation in each modified sentence. Data augmentation is applied to all data, resulting in a total of 2,284 data points generated after augmentation. After the data augmentation process is complete, an overlap data removal process is conducted to ensure that there is no identical or overlapping data between the training data and the test data. The results of the overlap data removal process show that the amount of training data remains at 1,826 data points, while the amount of test data decreases from 458 to 362 data points. This indicates that 96 data points in the test set have similarities with the training set. Consequently, these data points were removed to ensure the testing process is conducted on data that has never been seen by the model during training. The evaluation results in this scenario are presented in Table 7 below.

Table 7. Feature Extraction Test Results with Data Augmentation Technique

No.	Feature Extraction Method	Technique	Class	Recall	Precision	F1-Score	Accuracy
1	**TF-IDF**	**Augmentation**	**Negative**	**96%**	**96%**	**96%**	**94%**
			Neutral	**91%**	**100%**	**95%**	
			Positive	**92%**	**91%**	**92%**	

No.	Feature Extraction Method	Technique	Class	Recall	Precision	F1-Score	Accuracy
2	**BM25**	Non-Augmentation	Negative	83%	74%	78%	70%
			Neutral	14%	33%	20%	
			Positive	55%	63%	59%	
		Augmentation	**Negative**	**97%**	**98%**	**97%**	**97%**
			Neutral	**100%**	**100%**	**100%**	
			Positive	**96%**	**94%**	**95%**	
		Non-Augmentation	Negative	86%	79%	82%	74%
			Neutral	14%	20%	17%	
			Positive	64%	71%	67%	

Table 7 shows the performance results of the TF-IDF and BM25 feature extraction methods with synonym replacement-based data augmentation technique. The accuracy of both methods with the data augmentation technique increased significantly compared to the previous two scenarios, especially in the neutral class, which was previously difficult to classify effectively. In this scenario, the percentage of model classification on neutral class metric recall reached 91% for TF-IDF and 100% for BM25, while in the previous two scenarios, the percentage on neutral class metric recall only reached 14% and 21%. From these results, data augmentation proved to be more effective in improving the quality of data representation and enabling the model to understand review patterns across all classes. In other words, data augmentation is more effective than SMOTE in handling data imbalance and providing some data variation. This is because data augmentation creates a greater variety of the original data, especially for minority classes. By using the Synonym Replacement method, data augmentation generates new sentences with meaningful word changes without altering the original meaning, thus helping the model understand broader patterns [26]. The augmentation process not only increased recall but also reduced misclassification in ambiguous reviews. This confirms that synonym replacement significantly enriches the quality of training data without introducing semantic noise, outperforming oversampling-based approaches.

In this regard, Table 8 presents the evaluation metrics used to compare the performance of the TF-IDF and BM25 weighting methods across the three scenarios described earlier:

Table 8. Evaluation Metrics of All Scenarios

No.	Feature Extraction Method	Recall	Precision	F1-Score	Accuracy
1	TF-IDF	51%	50%	52%	70%
2	BM25	55%	57%	55%	74%
3	TF-IDF+SMOTE	51%	50%	51%	65%
4	BM25+SMOTE	52%	53%	52%	70%
5	TF-IDF+Augmentation	93%	96%	94%	94%
6	**BM25+ Augmentation**	**98%**	**98%**	**98%**	**97%**

From Table 8, it can be seen that in the scenario without additional techniques, the BM25 method shows superiority over TF-IDF, with accuracies of 74% and 70%, respectively. When the SMOTE technique is applied, the performance of both methods decreases slightly, indicating that SMOTE is less effective in improving the classification quality, especially for neutral classes where the amount of data is limited. This is because SMOTE only generates synthetic samples by interpolating existing minority data, without adding contextual diversity to the data [26]. In contrast, the Synonym Replacement-based data augmentation technique significantly improves model performance, with accuracy reaching 94% for TF-IDF and 97% for BM25. This technique not only increases the amount of data, but also enriches the semantic variation in the data, especially in minority classes, thus helping the model understand sentence patterns and context better. Thus, data augmentation proved to be more effective than SMOTE in overcoming data imbalance and improving the generalization ability of the model. Compared to existing literature using Naïve Bayes and SVM, our proposed method demonstrates higher precision and recall, especially in neutral sentiment classification, a known challenge in prior research. The accuracy improvement from 88% (Naïve Bayes) to 97% (MLP + BM25 + Augmentation) underlines the strength of our integrated approach.

By using the best model in the third scenario that employs BM25 feature extraction and data augmentation, the model is then used to test several reviews in the application as shown in Figure 3 below.

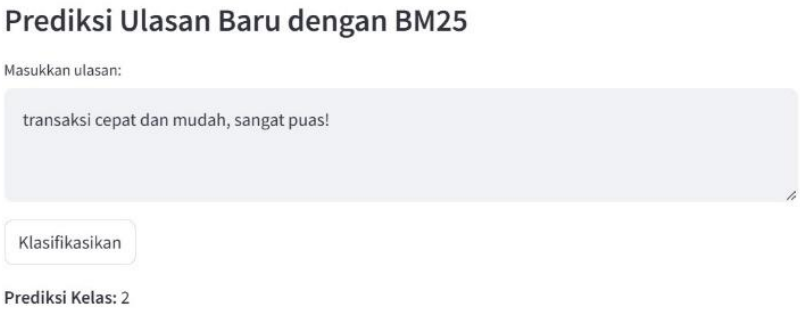


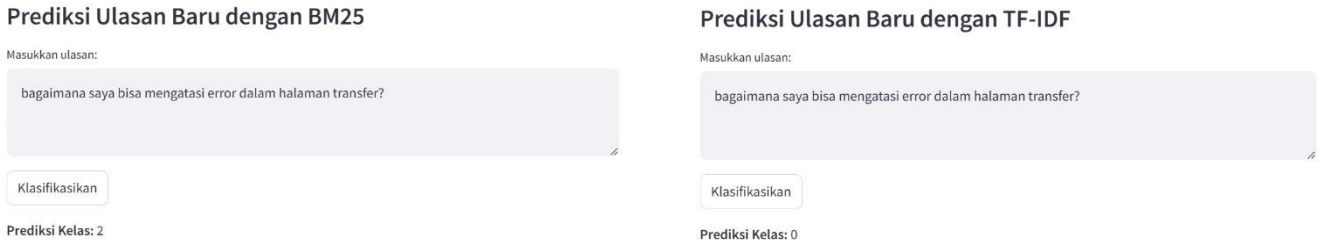
Figure 3. Model Testing Application on New Reviews

In Figure 3, there is an example of a review typed by a user in the form of '*transaksi cepat dan mudah, sangat puas!'* which is successfully classified as class 2, indicating that the review is classified as positive. This process illustrates how the classification model is used to assess the sentiment of Livin' by Mandiri app user reviews. In addition, some examples of reviews tested through the app is presented in Table 9.

Table 9. Streamlit Test Results on New Reviews

No.	Reviews	Labeling Result
1	*Fitur-fitur pada aplikasi livin sangat mudah digunakan*	2
2	*Di mana kantor livin?*	2
3	*Lambat dan sering tidak responsif, mohon diperbaiki.*	0
4	*Sering kali aplikasi mengalami crash saat saya ingin melakukan transfer, sangat mengganggu dan membuat saya tidak nyaman menggunakannya*	0
5	*Halo kak kenapa ya sejak semalam saya coba buat install livin yang baru tidak bisa*	1
6	*Kenapa livin terbaru pada saat pembayaran sudah ga bisa pilih pembayaran melalui debit/kredit ya?*	1

The test results show that the BM25 feature weighting technique consistently demonstrate superior results compared to the TF-IDF feature weighting technique, especially when combined with data augmentation techniques. For example, in handling new reviews such as "*bagaimana saya bisa mengatasi error dalam halaman transfer*?", BM25 was able to identify the review as a positive class, while TF-IDF classified it as a negative class, as shown in Figure 4 below.

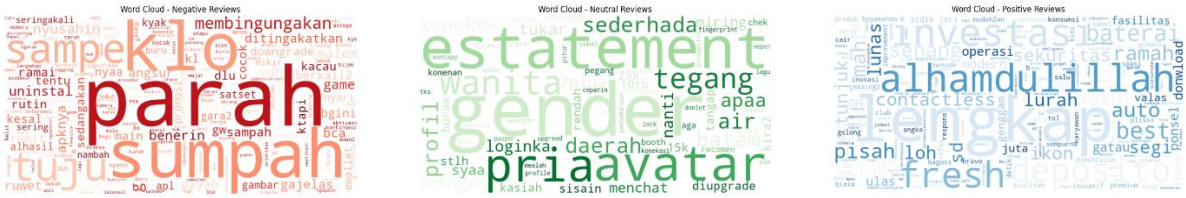


(a) (b)

Figure 4. Model Testing on (a) BM25 and (b) TF-IDF Feature Extraction

Based on Figure 4, the review stating "*bagaimana saya bisa mengatasi error dalam halaman transfer?*" shows that it does not contain negative elements, but only a question asked by the user. However, TF-IDF classifies the review in the negative class because it focuses solely on word frequency without taking the overall context into account [18]. Meanwhile, BM25 recognizes the sentence as a positive class because BM25 considers document length and keyword relevance using a logarithmic saturation function, making it more effective in understanding the context of short reviews [17].

To provide a deeper understanding of the word distribution in each review class, word cloud visualization can be used for each review class, as shown in Figure 5 below.



(a) (b) ©

Figure 5. (a) Negative, (b) Neutral, (c) Positive Class Wordcloud

Figure 5 present the word cloud visualizations for each sentiment class: negative, neutral and positive. Negative reviews are dominated by words like "*parah*", "*sumpah*" and "*membingungkan,*" which reflect disappointment or complaints. Neutral reviews feature words like "statement", "gender" and "avatar," which are informative without emotional content. Meanwhile, positive reviews contain words such as "*alhamdulillah*", "fresh" and "*mantap,*" which show user appreciation and satisfaction. In other words, the word cloud above helps identify dominant words and evaluates the extent to which the algorithm captures the context of the review.

4. Conclusion

Based on the research results, it can be concluded that the Multi-Layer Perceptron (MLP) model is able to process user review data from Livin' by Mandiri application with varying performance, depending on the data processing scenario used. The BM25 feature extraction method consistently yields better results than TF-IDF in all scenarios. This is due to the advantages of BM25, which considers document length and uses a logarithmic saturation function to calculate word weights, thereby avoiding the overweighting frequently occurring words. The application of data augmentation techniques also proved to significantly contribute to improving classification accuracy. This technique helps to balance the distribution between classes, including the neutral class, which previously showed low performance. The combination of BM25 feature extraction and data augmentation techniques produced the best performance, with accuracy reaching 97%, recall 98%, precision 98%, and F1-score 98%. These results show that the choice of feature extraction method plays an important role in determining the accuracy of the model. By selecting the right feature extraction strategy and applying appropriate augmentation techniques, the performance of the MLP-based classification model can be significantly improved in analyzing the sentiment of app user reviews. In addition, the application of this model provides deep insights for developers to understand user sentiment and improve app features, which, in turn, has the potential to increase customer satisfaction, user loyalty, and Bank Mandiri's reputation for providing more responsive and quality digital banking services. For future research, it is recommended to explore more complex classification models, such as Long Short-Term Memory (LSTM) or Transformer-based models like BERT, which have a better ability to understand context and word order.

References

[1] S. I. Murpratiwi, S. E. Anjarwani, I. G. P. S. Wijaya, and A. Aranta, "Sosialisasi Internet Sehat Dan Pelatihan Penggunaan Internet Sebagai Media Penunjang Pembelajaran Di SD Negeri Anggaraksa," *J. Begawe Teknol. Inf.*, vol. 4, no. 2, pp. 256–263, 2023. https://doi.org/10.29303/jbegati.v4i2.1111

[2] D. D. Audiansyah, D. E. Ratnawati, and B. T. Hanggara, "Analisis Sentimen Aplikasi MyXL menggunakan Metode Support Vector Machine berdasarkan Ulasan Pengguna di Google Play Store," vol. 6, no. 8, pp. 3987–3994, 2022.

[3] A. Ahmad, W. Gata, and S. Panggabean, "Sentimen Analisis dengan Long Short-Term Memory dan Synthetic Minority Over Sampling Technic Pada Aplikasi Digital Perbankan," *J. Teknol. Inf. dan Komun.*, vol. 8, no. 4, pp. 973–984, 2024.

[4] T. Wahudi and Z. Hutabarat, "Faktor-Faktor Yang Mempengaruhi Niat Penggunaan Digital Banking: Livin ' By Mandiri," *J. Tek. Inform. dan Sist. Inf.*, vol. 10, no. 1, pp. 509–525, 2023.

[5] S. H. Alviyanti, A. Purwandira, I. Febiyanti, E. Daniati, and A. Ristyawan, "Klasifikasi Sentimen Pengguna Aplikasi Livin By Mandiri Pada Playstore Menggunakan Algoritma Naive Bayes," *Agustus*, vol. 8, pp. 2549–7952, 2024.

[6] S. L. Ranataru and N. Trianasari, "Analisis Sentimen Media Sosial Terhadap Aplikasi Perbankan untuk Mengetahui Kepuasan Pengguna Aplikasi: Studi Kasus pada Livin by Mandiri dan BCA Mobile," *Al-Kharaj J. Ekon. , Keuang. Bisnis Syariah*, vol. 6, pp. 6818–6838, 2024. https://doi.org/10.47467/alkharaj.v6i9.3805

[7] I. D. Onantya and P. P. Adikara, "Analisis Sentimen Pada Ulasan Aplikasi BCA Mobile Menggunakan BM25 Dan Improved K-Nearest Neighbor," *J-Ptiik.Ub.Ac.Id*, vol. 3, no. 3, pp. 2575–2580, 2019.

[8] M. Z. Hariansyah and Siswanto, "Implementasi Metode Multinomial Naive Bayes pada Analisis Sentimen Terhadap Layanan Aplikasi Livin by Mandiri," *Semin. Nas. Mhs. Fak. Teknol. Inf.*, vol. 1, no. 1, pp. 517–524, 2022.

[9] N. Nurfadila, M. Ariyanti, and N. Trianasari, "Analisis Kualitas Layanan Mobile Banking New Livin' By Mandiri Menggunakan Sentiment Analysis," *JIBR J. Indones. Bus. Res.*, vol. 1, no. 1, pp. 77–82, 2023.

[10] C. A. Qurniaty and K. Kusnawi, "Ekspresi Emosi Berdasarkan Suara Menggunakan Algortima Multi Layer Perceptron dan Support Vector Machine," *Indones. J. Comput. Sci.*, vol. 12, no. 6, pp. 4014–4025, 2023. https://doi.org/10.33022/ijcs.v12i6.3567

[11] G. G. Warow and H. Pandia, "Analisis Sentimen Aplikasi Dana Menggunakan Naïve Bayes Classifier dan Support Vector Machine," *Jutisi J. Ilm. Tek. Inform. dan Sist. Inf.*, vol. 13, no. 1, p. 609, 2024. https://doi.org/10.35889/jutisi.v13i1.1893

[12] N. Meilani, Mhd. Furqan and Suhardi, "Analisis sentimen pengguna aplikasi BSI mobile akibat ransomware menggunakan algoritma support vector machine," *INFOTECH J. Inform. Teknol.*, vol. 5, no. 1, pp. 42–51, 2024. https://doi.org/10.37373/infotech.v5i1.1102

[13] R. A. Husen, R. Astuti, L. Marlia, R. Rahmaddeni, and L. Efrizoni, "Analisis Sentimen Opini Publik pada Twitter Terhadap Bank BSI Menggunakan Algoritma Machine Learning," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 3, no. 2, pp. 211–218, 2023. https://doi.org/10.57152/malcom.v3i2.901
[14] F. H. Pasaribu, N. Khairina, D. Noviandri, S. Susilawati, and R. Syah, "Analysis of The Multilayer Perceptron Algorithm on Twitter User's Sentiment Towards The COVID-19 Vaccine," *J. Informatics Telecommun. Eng.*, vol. 7, no. 1, pp. 155–163, 2023. https://doi.org/10.31289/jite.v7i1.9664
[15] L. Efrizoni, S. Defit, M. Tajuddin, and A. Anggrawan, "Komparasi Ekstraksi Fitur dalam Klasifikasi Teks Multilabel Menggunakan Algoritma Machine Learning," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 21, no. 3, pp. 653–666, 2022. https://doi.org/10.30812/matrik.v21i3.1851
[16] I. Daniel, A. Fahmi Limas Ptr, and A. Ichsan, "Klasifikasi Risiko Penyakit Serangan Jantung Dengan Multi-Layer Perceptron," *Data Sci. Indones.*, vol. 14, no. 1, pp. 57–64, 2024. https://doi.org/10.47709/dsi.v4i1.4667
[17] A. Purnamawati, M. N. Winarto, and M. Mailasari, "Analisis Sentimen Aplikasi TikTok menggunakan Metode BM25 dan Improved K-NN Fitur Chi-Square," *J. Komtika (Komputasi dan Inform.*, vol. 7, no. 1, pp. 97–105, 2023. https://doi.org/10.31603/komtika.v7i1.8938
[18] R. Maulana, A. Voutama, and T. Ridwan, "Analisis Sentimen Ulasan Aplikasi MyPertamina pada Google Play Store menggunakan Algoritma NBC," *J. Teknol. Terpadu*, vol. 9, no. 1, pp. 42–48, 2023. https://www.doi.org/10.54914/jtt.v9i1.609
[19] D. Duei Putri, G. F. Nama, and W. E. Sulistiono, "Analisis Sentimen Kinerja Dewan Perwakilan Rakyat (DPR) Pada Twitter Menggunakan Metode Naive Bayes Classifier," *J. Inform. dan Tek. Elektro Terap.*, vol. 10, no. 1, pp. 34–40, 2022. https://doi.org/10.23960/jitet.v10i1.2262
[20] N. Agustina, D. H. Citra, W. Purnama, C. Nisa, and A. R. Kurnia, "Implementasi Algoritma Naive Bayes untuk Analisis Sentimen Ulasan Shopee pada Google Play Store," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 2, no. 1, pp. 47–54, 2022. https://doi.org/10.57152/malcom.v2i1.195
[21] R. A. Nurdian, Mujib Ridwan, and Ahmad Yusuf, "Komparasi Metode SMOTE dan ADASYN dalam Meningkatkan Performa Klasifikasi Herregistrasi Mahasiswa Baru," *J. Tek. Inform. dan Sist. Inf.*, vol. 8, no. 1, pp. 24–32, 2022. https://doi.org/10.28932/jutisi.v8i1.4004
[22] S. Sasmita, R. N. Jariah S.Intam, D. F. Surianto, and M. F. B, "Analisis Sentimen Terhadap Kontroversi Putusan MK Mengenai Usia Capres-Cawapres Menggunakan Multi-Layer Perceptron Dengan Teknik SMOTE," *Fakt. Exacta*, vol. 17, no. 2, p. 188, 2024. https://doi.org/10.30998/faktorexacta.v17i2.22442
[23] W. Wahyudi, R. Kurniawan, and Y. Arie Wijaya, "Analisis Sentimen Pengguna Terhadap Aplikasi Blu Bca Di Playstore Mengunakan Algoritma Naïve Bayes," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 3, pp. 2511–2517, 2024. https://doi.org/10.36040/jati.v8i3.9216
[24] R. N. Harahap and K. Muslim, "Peningkatan Akurasi pada Prediksi Kepribadian Mbti Pengguna Twitter Menggunakan Augmentasi Data," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 4, p. 815, 2020. https://doi.org/10.25126/jtiik.2020743622
[25] A. Nur Azizah, M. Falach Asy'ari, I. Wisma Dwi Prastya, and D. Purwitasari, "Easy Data Augmentation untuk Data yang Imbalance pada Konsultasi Kesehatan Daring," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 5, pp. 1095–1104, 2023. https://doi.org/10.25126/jtiik.20231057082
[26] I. Athiyyah Rahma and L. Hulliyyatus Suadaa, "Penerapan Text Augmentation Untuk Mengatasi Data Yang Tidak Seimbang Pada Klasifikasi Teks Berbahasa Indonesia Studi Kasus: Deteksi Judul Clickbait Dan Komentar Hate Speech Pada Berita Online," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 6, pp. 1329–1340, 2023. https://doi.org/10.25126/jtiik.2023107325
[27] A. I. Tanggraeni and M. N. N. Sitokdana, "Analisis Sentimen Aplikasi E-Government pada Google Play Menggunakan Algoritma Naïve Bayes," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 9, no. 2, pp. 785–795, 2022. https://doi.org/10.35957/jatisi.v9i2.1835