P-ISSN: 2087-1244
E-ISSN: 2476-907X

Balinese Language Classification on Social Media using Multinomial Naive Bayes Method with TF-IDF

Putu Widyantara Artanta Wibawa[1]*; Cokorda Pramartha[2];
I Gusti Ngurah Anom Cahyadi Putra[3]; Luh Gede Astuti[4]

[1-4]Informatics, Faculty of Mathematics and Natural Sciences, Udayana University,
Denpasar, Indonesia, 80361
[1]putuwaw973@gmail.com; [2]cokorda@unud.ac.id;
[3]anom.cp@unud.ac.id; [4]lg.astuti@unud.ac.id

Received: 2nd August 2025/ **Revised:** 5th October 2025/ **Accepted:** 11th November 2025

How to Cite: Wibawa, P. W. A., Pramartha, C., Putra, I. G. N. A. C., & Astuti, L. G. (2025). Balinese language classification on social media using Multinomial Naive Bayes method with TF-IDF. *ComTech: Computer, Mathematics and Engineering Applications, 17*(1), 23−37. https://doi.org/10.21512/comtech.v17i1.14132

IN PRESS

***Abstract* -** Balinese is a local language widely used and spoken by Balinese people, including on social media. However, the nuances of these politeness levels are often lost in informal digital communication, and there is a significant lack of computational models to classify them, especially in low-resource languages like Balinese, automatically. The primary objective of this study is to evaluate the performance of the Multinomial Naive Bayes method combined with Term Frequency-Inverse Document Frequency (TF-IDF) feature extraction, Chi-square feature selection, and Synthetic Minority Oversampling Technique (SMOTE) in classifying Balinese language levels. The dataset for this study consists of 1,314 annotated social media posts and comments, primarily sourced from Instagram. A Balinese language expert conducted the annotation, categorizing the text into six levels that represent varying degrees of politeness and formality. These levels are alus singgih (polite, used for respecting others), alus sor (polite, used for self-humbling), alus mider (polite, used for both respecting others and self-humbling), alus madia (an intermediate level of politeness), basa andap (casual, commonly used in everyday life), and basa kasar (impolite, often used during arguments or toward animals). The experimental results showed that the model achieved 96.53% accuracy on the training data and 61.45% on the test data. Additionally, hyperparameter tuning revealed that the Multinomial Naive Bayes model with 2,720 selected features and SMOTE oversampling achieved 91.78% accuracy, significantly outperforming the baseline model without feature selection or oversampling, which achieved only 64.93% accuracy.

***Keywords*:** classification, Balinese language, TF-IDF, Chi-square, SMOTE, Multinomial Naive Bayes

I. INTRODUCTION

Balinese is a local language widely spoken by Balinese people. Currently, rapid technological development has changed people's lifestyles, including how they communicate through social media. According to Mastini et al. (2021), the existence of social media also provides a new space for people to post about the ongoing use of the Balinese language. However, when using social media, people who use Balinese tend not to use the appropriate level of language (anggah ungguhing basa). This is very important because anggah ungguhing basa is a speech-level system in the Balinese language based on varying degrees of politeness and formality, which reflects a person's ethics and influences politeness in communication (Suardiana, 2019). Some are used to show respect to others; others are more casual for everyday use, including language that can be impolite and should be used with care because it can be offensive in certain contexts. For example, in Balinese, there are more than four words to express the meaning of "death". The choice of words depends on the social status and the person being referred to. Using an inappropriate term, such as a casual or low-level expression for an elder, would be considered disrespectful and socially unacceptable within the community.

Given the importance of properly using the Balinese language level in communication, especially

*Corresponding Author

on social media, there is a need for an automated system to classify these levels. For instance, in social media content moderation to detect inappropriate or offensive language, or in digital writing assistants that suggest more suitable speech levels to ensure respectful, appropriate communication. One classification method is Multinomial Naive Bayes. Multinomial Naive Bayes was chosen because it is widely recognized as a state-of-the-art baseline for text classification (Raza et al., 2025). Furthermore, Multinomial Naive Bayes is suitable for low-resource languages, such as Balinese, where the available dataset is relatively small. This method works with Bayes' theorem and term frequency. However, terms or words that appear too often in a document can make it less meaningful. One solution to overcome this is to use TF-IDF (Term Frequency-Inverse Document Frequency) feature extraction, which can determine how important a term is in a document. In text classification, features that are too large can reduce performance (Hamzah, 2021); therefore, a feature selection process is needed, one of which is Chi-square feature selection.

In text classification, several studies use the Multinomial Naive Bayes method because it is simple, fast, and easy to apply. One study that applies Multinomial Naive Bayes is Ardhana's (2018) classification of language levels in Javanese articles, using Multinomial Naive Bayes with N-Gram and TF-IDF feature extraction and the SMOTE (Synthetic Minority Over-Sampling Technique) resampling method. This study reported precision, recall, and accuracy of 72.67%, 75.00%, and 74.99%, respectively, when using TF-IDF and SMOTE, with TF-IDF feature extraction yielding better results than N-Gram.

Another study by Angeline et al. (2022) used the Multinomial Naive Bayes method to classify dialects in Java. This study used TF (Term Frequency) feature extraction and the SMOTE method to determine the effect of oversampling techniques on algorithm performance. This study achieved the best performance, with 96.97% accuracy, 97.53% precision, and 96.83% recall. Another study that used the Naive Bayes method, combined with rule-based and N-Gram stemming, to detect hate speech was conducted by Dewi & Putra (2021), which achieved an accuracy of 85%. Research conducted by Azad et al. (2021) on the Kurdish language also implemented TF-IDF as the feature selection method.

This study adds novelty by applying the Multinomial Naive Bayes method with TF-IDF feature extraction, the SMOTE oversampling technique, and Chi-square feature selection to classify Balinese language levels. The stages carried out in this study include dataset collection, text preprocessing, feature extraction with TF-IDF, feature selection with Chi-square, application of SMOTE oversampling, searching for the best hyperparameters with hyperparameter tuning, training the Multinomial Naive Bayes model with the best parameters, and finally model evaluation using a confusion matrix and an accuracy score.

II. METHODS

This study was conducted in several stages, as shown in Figure 1 (see Appendices). The first stage is data collection. The data used in this study are primary data collected through scraping of posts and comments on social media, especially Instagram. The dataset obtained will later be stored in JSON (JavaScript Object Notation) format. This format is used because JSON is quite easy to store data, especially when the data has many nested attributes. The attributes from the scraping results that will be stored are the post text and the post's URL (Uniform Resource Locator) or source.

The dataset will then be filtered to the last 5 years of data, excluding data that is dominated by or contains more than half of its content in Balinese, and selecting data with a minimum length of 5 words. Then, the dataset will be given to Balinese language experts, in this case, *Penyuluh Bahasa Bali* and their team, who will annotate or categorize it based on the appropriate speech levels. *Penyuluh Bahasa Bali* is a group of professionals assigned by the Cultural Department of Bali Province to carry out activities aimed at preserving, fostering, and developing the Balinese language, script, and literature. Finally, the annotated dataset from the Balinese language expert will be tidied to ensure uniform labels. All stages of data collection are shown in Figure 2 (see Appendices).

Since this study focuses on classification, there are labels used. In this study, labels are used in the form of Balinese language levels. According to Suwija (2019), the Balinese language can be classified into six levels based on the words used in a sentence. Therefore, in this study, six levels of Balinese language are used, namely *alus singgih* (polite, used for respecting others), *alus sor* (polite, used for self-humbling), *alus mider* (polite, used for both respecting others and self-humbling), *alus madya* (an intermediate level of politeness), *basa andap* (casual, commonly used in everyday life), and *basa kasar* (impolite, often used during arguments or toward animals).

Basa alus singgih is a Balinese term of respect, used to address people of higher rank. *Basa alus sor* is a Balinese language with a polite meaning, used to humble oneself (Suwija, 2019). *Basa alus mider* is a Balinese language phrase with a polite meaning, used to both respect others and humble oneself. *Basa madia* is a level of Balinese language classified as intermediate, with a language value between *andap* and *alus*. *Basa andap* is a level of the Balinese language commonly used in everyday life (Sosiawan et al., 2021). Finally, *basa kasar* is a Balinese language with a bad-taste value, is very impolite, and is often used during arguments or directed at animals.

The next stage is text preprocessing, which converts the original text data into structured data and identifies the most significant text features for distinguishing between text categories (Hickman et al., 2022). Before the data is processed for the training stage, it will be prepared during the text preprocessing

IN PRESS

stage. The processes carried out during the preprocessing stage are case folding, removal of non-alphabet characters, removal of stop words, stemming, and tokenization. The stemming technique used in this study is a rule-based approach grounded in the structure of the Balinese language. A study conducted by Nugraha & Wardani (2020) also employed a similar rule-based approach. Likewise, Agus et al. (2019) applied a comparable method by combining rule-based stemming with an N-gram approach.

Next is feature extraction with TF-IDF. In this stage, a matrix will be generated with the number of rows equal to the number of datasets and the number of columns (features) equal to the number of different tokens in the dataset. The main idea is that if a word or phrase appears frequently in an article and is rarely found in other articles, it is considered that the word or phrase has good class discrimination ability and suitability for classification (Gifari et al., 2022). The mathematical representation of the term weight in a document by TF-IDF is shown by Equation (1):

$$W(d,t) = tf(d,t) \times log\left(\frac{N}{df(t)}\right) \qquad (1)$$

Where *d* is document *d, t*, *t* is term *t*, *tf*(*d, t*) is the number of terms *t* in document *d, n* is the number of documents, and *df*(*t*) is the number of documents containing term *t* in the corpus (Zhou, 2022). After the features are successfully extracted, the next step is to perform feature selection using the Chi-square test. In this stage, features with high relevance or importance will be selected. Because the number of features obtained is not yet known, the percentage of all features will be used. Compared with several other feature selection methods, such as Information Gain (IG), Mutual Information (MI), and Gini Coefficient (GI), Chi-Square is among the most effective feature selection algorithms (Bahassine et al., 2020). The chi-square statistic measures the association between feature or word *t* and class *c*. The distribution between word *t* and class *c* is shown in Table 1.

Table 1 Feature and Category

	c	¬c	Total
t	*A*	*B*	*A*+*B*
¬t	*C*	*D*	*C*+*D*
Total	*A*+*C*	*B*+*D*	*N*

It is assumed that feature *t* and class *c* conform to a Chi-square distribution with first-order degrees of freedom. The higher the Chi-square score for class *c*, the more category information feature *t* carries, and the greater the relevance between *t* and *c*. The formula of Chi-square for feature *t* with class *c* is shown in Equation (2) (Bahassine et al., 2020)

$$\chi^2(t,c) = \frac{N(AD-CB)^2}{(A+C)(B+D)(A+B)(C+D)} \qquad (2)$$

After the important features are successfully selected, the next step is to perform SMOTE oversampling. Oversampling is performed using the non-majority approach, meaning that all minority or non-dominant classes are oversampled until their numbers match those of the majority class, and no minority classes remain. In other words, the oversampling stage will make all classes have the same distribution or number (Chen et al., 2024)

After oversampling, the next step is to perform hyperparameter tuning to find the best hyperparameters for training the Multinomial Naive Bayes model. One way to perform hyperparameter tuning is to use grid search to find a combination of predetermined hyperparameter values (Hossain & Timmer, 2021). The hyperparameters to be searched are alpha in Multinomial Naive Bayes (which serves as Laplacian smoothing); the number of features selected via the Chi-square method; and the application of SMOTE oversampling, as specified in Table 2.

Table 2 Hyperparameter Options

No	Parameter	Options
1.	Alpha	0.25, 0.5, 0.75, 1
2.	% Features	40%, 50%, 60%, 70%, 80%, 90%, 100%
3.	SMOTE	True, False

The hyperparameter tuning scheme is K-fold cross-validation with a fixed K of 5 (Nti et al., 2021). This scheme works by dividing the dataset into two parts: training and test data. In this study, the training data will be split into 80%, with the remaining 20% used for testing. Once the best hyperparameters are obtained, the model will be retrained with them. The stages of this hyperparameter tuning are shown in Figure 3 (see Appendices).

In the training stage, the model used is Multinomial Naive Bayes, a probabilistic method that focuses on text classification (Zulfikar et al., 2023). In principle, Multinomial Naive Bayes calculates the probability that each word in a sentence belongs to each class using prior and likelihood probabilities, according to Bayes' theorem. As for the Multinomial Naive Bayes formula used with TF-IDF weighting, it is shown by Equation (3) (Damanik & Setyohadi, 2021)

$$P(X_n|C) = \frac{\sum tf(X_n, d \in C) + \alpha}{\sum N_{d\in C} + V} \qquad (3)$$

Where $\sum tf(X_n, d\epsilon C)$ is the weighted sum of the words X_n from all documents in the training data in category *C*. Meanwhile, $\sum N_{d\epsilon C}$ is the weighted sum of all terms in the training data in category *C*. Here, *α* is the Laplace smoothing parameter, and *V* is the total

IN PRESS

number of words or tokens in the training data.

To ensure end users can effectively use the system, an interface will be developed as a simple website. The development process uses a prototyping model, an enhancement of the SDLC (Software Development Life Cycle) method. Unlike the traditional waterfall model—often criticized for its lengthy stages and high costs (Made et al., 2022)—prototyping offers a more iterative and flexible approach.

As shown in Figure 4 (see Appendices), the prototyping process begins with requirements gathering, which includes identifying both functional and non-functional system requirements. The next phase involves building a prototype or mid-fidelity prototype that serves as the initial interface design (Pramartha et al., 2023). This prototype is then evaluated to assess whether it meets the system requirements. If the evaluation indicates shortcomings, the process loops back—either to the prototype design phase for refinement or to the requirements-gathering phase for further input. Once the prototype satisfies the requirements, the system proceeds to the development phase, where it is implemented in code.

In the final evaluation stage, system performance is assessed using a confusion matrix, which provides a detailed view of prediction results in terms of TP (True Positives), TN (True Negatives), FP (False Positives), and FN (False Negatives) (Valero-Carreras et al., 2023). In addition to accuracy metrics, the evaluation also examines the influence of specific parameters—such as the alpha value, the SMOTE oversampling technique, and Chi-square feature selection—on the classification model's performance. This study particularly analyzes how these combinations impact the accuracy of the Multinomial Naive Bayes classifier.

III. RESULTS AND DISCUSSIONS

The data in this study were obtained through web scraping techniques, specifically by extracting text from captions and comments on selected Instagram posts. The data were collected from several Instagram accounts, including @lanmalajah.id, @ basabali.id, @basabaline, and @melajahbahasabali, between October 29 and November 7, 2024. These accounts were chosen because they frequently post content in Balinese and have substantial followings, making them likely sources of rich, relevant linguistic data.

A targeted, account-based scraping approach was adopted to overcome the limitations of keyword-based searches, which often yield inaccurate or irrelevant results on Instagram. The results of scraping are still raw data stored in JSON format. The information includes the post URL, caption, and comments. The combined caption and comment data obtained is 18,224. An example of the raw dataset is shown in Figure 5 (see Appendices).

The next stage is to filter the data. From 18,224 data points, the data that predominantly contains Balinese will be selected. Dominant is defined as text that contains Balinese words more than half of the total words. Non-Balinese words that appear in these entries were retained as long as the text met the dominant Balinese threshold. With this filter, 2,790 data points were obtained. The last stage is to filter for data with more than 5 words; from this stage, the final dataset contains 1,350 data points. The filtered dataset will then be annotated by 4 experts, including a Balinese language expert, according to the Balinese language level. A sample of the annotated dataset is shown in Table 3.

IN PRESS

Table 3 Sample Data with Annotation

No.	Text (Balinese)	Translate (English)	Label (Language Level)
1	*Sekadi patut pisan nike kakak cantik…milet pisan titiang wikan sekadi kakak*	That’s absolutely right, beautiful sister. I really want to be as smart as you.	*Alus Singgih*
2	*Becik pisan satua ne gek*	The story she told is very good.	*Alus Sor*
3	*Yening krame Banjar Boye je semeton Jero. sepatutne Ida dane sane patut.ampura*	If speaking with community members (*krama banjar*), do not use “*semeton*” or “*jero*”; instead, use “*ida dane*”.	*Alus mider*
4	*Becik pisan tatanan wacananne. Untuk memelihara kauningane puniki becikne lanturang ngripta pustaka wangun gancaran utawi irisan/ puisi.*	The structure of the text is very good. To retain your knowledge, you should continue creating other forms, such as prose (*gancaran*) or poetry.	*Alus Madia*
5	*Suud mebalih ne, Mare tiang nawang ternyata basa bali nika keweh*	After watching this, I realized Balinese is really hard.	*Basa Andap*
6	*Lengeh ci,ngoyong lengeh oyongan iban ci dikubu*	You’re stupid, just stay in your little house.	*Basa Kasar*

Before text preprocessing, label standardization will be carried out to ensure uniform language levels. Label standardization is carried out because several labels given by the extension worker need to be tidied up, for example*, 'alus mider*' and *'Alus Mider*'. There are spelling errors, such as *'alis singgih*,' which should be *'Alus Singgih*'. After being normalized into 6 classes, namely *alus singgih*, *alus sor*, *alus mider*, *alus madia*, *basa andap*, and *basa kasar*, a dataset of 1,314 data was obtained with the distribution shown in Figure 6 (see Appendices).

From the distribution in Figure 6 (see Appendices), it can be seen that the data distribution is quite unbalanced, as the class or label' *basa kasar*' contains only 12 data, or 0.9% of the total dataset. Meanwhile, data labeled *basa andap* is quite large, totaling 549 records, or around 41.8% of the total dataset. There are very few *basa kasar* because it is rarely used, especially on social media, where it is inappropriate in the public domain, and on non-verbal media, it can make people look rude. Next, the dataset of 1,314 data will be divided into 80% training data and 20% test data, yielding 1052 training data and 262 test data. The distribution of training data is shown in Figure 7 (see Appendices).

Meanwhile, the distribution of test data classes is shown in Figure 8 (see Appendices). Both the training and test data show an unbalanced distribution. To address this issue, the training data will be resampled using SMOTE, and the performance will be evaluated during the hyperparameter tuning. The test data will remain unaltered, as it is intended to represent real-world conditions.

The existing training data will then be used for cross-validation and hyperparameter tuning to determine parameter values that yield the best performance. The parameters to be used are in accordance with Table 2. The options for alpha values are 0.25, 0.5, 0.75, and 1. While the number of features used is selected using the Chi-square feature selection, with options of 40%, 50%, 60%, 70%, 80%, 90%, and 100% of the total features, respectively equal to 1360, 1700, 2040, 2380, 2720, 3060, and 3400 features. Finally, whether SMOTE oversampling is used, the oversampling scheme is not the majority, which oversamples all classes that are not the majority.

With these parameters, a total of 56 parameter combinations were obtained. Furthermore, the process of finding the best parameters involved training the Multinomial Naive Bayes model on the training data using cross-validation, with performance evaluated using accuracy metrics. After hyperparameter tuning, the results of the comparison, shown in Figure 9 (see Appendices) as boxplots of accuracy between datasets treated with oversampling and those not, are presented. The results show that the oversampled dataset had a higher average accuracy.

Figure 10 (see Appendices) shows a comparison of average accuracy results as a function of the number of features and alpha during hyperparameter tuning using SMOTE oversampling. It can be seen that even with only a very small number of features, 40% is not enough to achieve the model's best performance. However, using all the features does not guarantee the best model performance.

Figure 11 (see Appendices) shows the relationship between the number of features and alpha and the average accuracy during hyperparameter tuning on a dataset without SMOTE oversampling. The graph also shows that choosing the right alpha and the number of features affects the model's performance. However, without oversampling, the performance is much lower than with SMOTE. Based on the results in Figures 10 (see Appendices) and 11 (see Appendices), it is necessary to select the appropriate number of features and hyperparameter values to achieve the highest average accuracy. Therefore, the model with the best accuracy of 91.78% in Figure 10 (see Appendices) is obtained, trained using the parameters shown in Table 4.

From Table 4, the best parameters are 80% of the total number of features (2720), the highest Chi-square value, alpha = 0.25, and SMOTE oversampling. Once it is determined that SMOTE oversampling is the best parameter, the training data will be oversampled, yielding results as shown in Figure 12 (see Appendices). The results of oversampling with the not-majority strategy will sample all classes that are not the majority class; in this case, not the *basa andap* class. Then, continue by retraining the Multinomial Naive Bayes model with the best parameters.

IN PRESS

Table 4 Best Parameter from Hyperparameter Tuning

Parameter	**Value**
Alpha	0.25
Features	2720
% Features	80%
SMOTE	True

Based on Table 5, the model trained with the best parameters achieved an accuracy of 96.53%. Overall, the model performed well across all classes, particularly for the *basa kasar* class, achieving perfect recall (1.0) and the highest F1-score of 0.9955. This indicates that the *basa kasar* class has distinct features and is easily recognized by the model. Meanwhile, the *alus mider* class had the lowest scores, indicating that the model had more difficulty classifying. This finding is consistent with the fact that most words from *alus mider* can be used in *alus singgih* and *alus sor.* For example, words such as *semeton* (friends, relatives), *bali* (Bali), *wenten* (there is), *sampun* (already), *ring* (at, on, in), *puniki* (this), and *sane* (that, which) are among the most common in *alus singgih* and *alus sor*, thus creating greater ambiguity

The confusion matrix in Figure 13 (see Appendices) shows that the model effectively learned the training data, as indicated by the strong concentration of colors along the main diagonal. However, the matrix also reveals the model's difficulty in classifying the alus *mider* class. It misclassified 19 instances as *alus madia* and 13 instances as the *alus sor* class.

Based on the testing data results in Table 6, the model achieved an accuracy of only 61.45%. The precision and recall values in the table show that while the model is proficient at classifying the *basa andap* class. However, it struggles to categorize the *alus sor* and *alus singgih* classes.

The confusion matrix for the test data in Figure 14 (see Appendices) shows that a significant number of *alus singgih* instances were misclassified as *alus sor* and *alus mider*. Additionally, numerous instances from the *alus mider* class were incorrectly classified as *alus madia*. This indicates that the model still struggles to distinguish between these closely related classes.

Based on the existing results, the model still cannot generalize effectively on the test data, even though it performs well on the training data augmented with SMOTE. This behavior indicates overfitting, as evidenced by the large gap between the training and test accuracies. One reason is that the test dataset is unbalanced, and oversampling cannot be applied to it because it must remain unseen and reflect the actual data distribution. Furthermore, because the number of some minority classes in the test data is very small, the model has difficulty making accurate predictions when new words appear that are not in the training data. In addition, most of the common words, such as *sampun*, *bali*, *semeton*, *puniki*, *sane*, *wenten*, and *ring*, frequently appear across *alus singgih*, *alus sor*, *alus mider*, and *alus madia*. This suggests that the distinction between these levels does not lie in the common vocabulary but rather in the subtle use of specific, less frequent discriminative words, which the model struggles to differentiate between the classes.

After model evaluation, the model will be integrated into the end-user system, allowing users to classify Balinese language levels easily. The prototype of the end-user system is shown in Figure 15 (see Appendices). After entering a sentence, the user can view the predicted level of that sentence along with detailed prediction results, including the probability for each class.

IN PRESS

Table 5 Evaluation Results on Training Data

Class	Precision	Recall	F1-Score	Accuracy
Alus Singgih	0.9664	0.9751	0.9707	0.9653
Alus Sor	0.9564	0.9932	0.9745	
Alus Mider	0.9416	0.9118	0.9264	
Alus Madia	0.9416	0.9842	0.9624	
Basa Andap	0.9976	0.9276	0.9613	
Basa Kasar	0.9910	1.0000	0.9955	

Table 6 Evaluation Results on Testing Data

Class	Precision	Recall	F1-Score	Accuracy
Alus Singgih	0.4118	0.4118	0.4118	0.6145
Alus Sor	0.3103	0.6000	0.4091	
Alus Mider	0.5968	0.5606	0.5781	
Alus Madia	0.3696	0.4595	0.4096	
Basa Andap	0.9432	0.7757	0.8513	
Basa Kasar	0.3333	0.3333	0.3333	

IV. CONCLUSIONS

This study aimed to address the challenge of automatically classifying the levels of Balinese text in social media. Using the Multinomial Naive Bayes method to classify 1314 data, the results show 96.53% accuracy on the training data and 61.45% on the test data. The optimal number of features, selected using Chi-square and oversampling with SMOTE during hyperparameter tuning, yields the best performance during training. The significant limitation identified during the experiment is overfitting due to a large gap between the training and test data. Despite the limitations, this study provides a machine learning approach for a low-resource language classification task. It produces a dataset from a baseline model that serves as a foundation for future research.

Based on the study and results, future work should prioritize expanding the datasets to achieve a more balanced class distribution, thereby improving the model's performance. In addition, explore more diverse data collection methods, as this study relied on data from only a limited number of accounts. This would allow the resulting dataset to be more varied and comprehensive, covering a broader range of topics. To potentially improve performance and better capture context, explore more advanced computational methods such as deep learning or Transformers.

AUTHOR CONTRIBUTIONS

Conceived and designed the analysis, P. W. A. W. and C. P.; Collected the data, P. W. A. W.; Contributed data or analysis tools, P. W. A. W. and C. P.; Performed the analysis, P. W. A. W.; Wrote the paper, P. W. A. W. and C. P.; Other contribution, I. G. N. A. C. P. and L. G. A.

DATA AVAILABILITY

The data that support the findings of this study are openly available in Zenodo at https://doi.org/10.5281/zenodo.17582015, reference number 17582015.

REFERENCES

Agus, M., Subali, P., & Fatichah, C. (2019). Kombinasi Metode Rule-Based dan N-Gram Stemming untuk Mengenali Stemmer Bahasa Bali. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *6*(2), 219–228. https://doi.org/10.25126/JTIIK.2019621105

Angeline, G., Wibawa, A. P., & Pujianto, U. (2022). Klasifikasi Dialek Bahasa Jawa Menggunakan Metode Naives Bayes. *Jurnal Mnemonic*, *5*(2), 103–110. https://doi.org/10.36040/mnemonic.v5i2.4748

Ardhana, A. P. (2018). *Klasifikasi Tingkatan Bahasa pada Artikel Berbahasa Jawa dengan Metode Multinomial Naïve Bayes*.

Azad, R., Mohammed, B., Mahmud, R., Zrar, L., & Sdiq, S. (2021). Fake News Detection in low-resourced languages "Kurdish language" using Machine learning algorithms. *Turkish Journal of Computer and Mathematics Education*, *12*(6), 4219–4225.

Bahassine, S., Madani, A., Al-Sarem, M., & Kissi, M. (2020). Feature selection using an improved Chi-square for Arabic text classification. *Journal of King Saud University - Computer and Information Sciences*, *32*(2), 225–231. https://doi.org/10.1016/J.JKSUCI.2018.05.010

Chen, W., Yang, K., Yu, Z., Shi, Y., & Chen, C. L. P. (2024). A survey on imbalanced learning: latest research, applications and future directions. *Artificial Intelligence Review*, *57*(6), 1–51. https://doi.org/10.1007/S10462-024-10759-6/FIGURES/11

Damanik, F. J., & Setyohadi, D. B. (2021). Analysis Of Public Sentiment About Covid-19 In Indonesia On Twitter Using Multinomial Naive Bayes And Support Vector Machine. *IOP Conference Series: Earth and Environmental Science*, *704*(1), 012027. https://doi.org/10.1088/1755-1315/704/1/012027

Dewi, D. A. E. R., & Putra, A. A. N. M. A. (2021). Kebencian Pada Bahasa Bali Dengan Metode Naive Bayes. *Jurnal Teknologi Informasi Dan Komputer*, *7*(2).

Gifari, O. I., Adha, M., Rifky Hendrawan, I., Freddy, F., & Durrand, S. (2022). Film Review Sentiment Analysis Using TF-IDF and Support Vector Machine. *Journal of Information Technology*, *2*(1), 36–40. https://doi.org/10.46229/JIFOTECH.V2I1.330

Hamzah, M. B. (2021). Classification of Movie Review Sentiment Analysis Using Chi-Square and Multinomial Naive Bayes with Adaptive Boosting. *Journal of Advances in Information Systems and Technology, 3*(1), 67–74. https://doi.org/10.15294/JAIST.V3I1.49098

Hickman,L.,Thapa,S.,Tay,L.,Cao,M.,&Srinivasan,P.(2022). Text Preprocessing for Text Mining in Organizational Research: Review and Recommendations. *Organizational Research Methods*, *25*(1), 114–146. https://doi.org/10. 1177/1094428120971683; Website:Website:Sage;Journal:Journal:Orma;Requestedjournal:Journal:Orma;Wgroup:String: Publication.

Hossain, R., & Timmer, D. (2021). Machine Learning Model Optimization with Hyper Parameter Tuning Approach. *Glob. J. Comput. Sci. Technol. D Neural Artif. Intell*, *21*(2).

Made, I., Wirawan, W., & Pramartha, C. (2022). PENGEMBANGAN SISTEM INFORMASI PENANGANAN PENDERITA GANGGUAN JIWA DENGAN PENDEKATAN ENTEPRISE SYSTEMS. *SINTECH (Science and Information Technology) Journal*, *5*(1), 31–41. https://doi.org/10.31598/SINTECHJOURNAL.V5I1.1070

Mastini, G. N., Kantriani, N. K., & Arini, N. W. (2021). Peran Media Sosial Instagram Dalam Upaya Menjaga Eksistensi Bahasa Bali. *Ganaya : Jurnal*

IN PRESS

Ilmu Sosial Dan Humaniora, *4*(2), 686–695. https://doi.org/10.37329/ganaya.v4i2.1414

Nti, I. K., Nyarko-Boateng, O., & Aning, J. (2021). Performance of Machine Learning Algorithms with Different K Values in K-fold CrossValidation. *International Journal of Information Technology and Computer Science*, *13*(6), 61–71. https://doi.org/10.5815/IJITCS.2021.06.05

Nugraha, P. G. S. C., & Wardani, N. W. (2020). STEMMING DOKUMEN TEKS BAHASA BALI DENGAN METODE RULE BASE APPROACH. *JATISI*, *7*(3), 510–521. https://doi.org/10.35957/JATISI.V7I3.538

Pramartha, C., Made, I., Mahendra, Y., Primahadi, G., Rajeg, W., & Arka, W. (2023). The Development of Semantic Dictionary Prototype for the Balinese Language. *International Journal of Cyber and IT Service Management (IJCITSM)*, *3*(2), 96–106. https://doi.org/10.34306/IJCITSM.V3I2.132

Sosiawan, P., Martha, I. N., & Artika, I. W. (2021). PENGGUNAAN BAHASA BALI PADA KELUARGA MUDA DI KOTA SINGARAJA. *Jurnal Pendidikan Dan Pembelajaran Bahasa Indonesia*, *10*(1), 40–54. https://doi.org/10.23887/JURNAL_BAHASA.V10I1.403

Raza, M. O., Mahoto, N. A., Shaikh, A., Pathan, N., Alshahrani, H., & Elmagzoub, M. A. (2025). A Machine Learning Approach of Text Classification for High- and Low-Resource Languages. *Computational Intelligence*, *41*(4), e70114. https://doi.org/10.1111/COIN.70114

Suardiana, I. W. (2019). Bahasa Bali dan Pemertahanan Kearifan Lokal. *Linguistika*, *19*(1), 1–7.

Suwija, I. (2019). Tingkat-Tingkatan Bicara Bahasa Bali (Dampak Anggah-Ungguh Kruna). *Sosiohumaniora*, *21*(1), 90. https://doi.org/10.24198/sosiohumaniora.v21i1.19507

Valero-Carreras, D., Alcaraz, J., & Landete, M. (2023). Comparing two SVM models through different metrics based on the confusion matrix. *Computers & Operations Research*, *152*, 106131. https://doi.org/10.1016/J.COR.2022.106131

Zhou, H. (2022). Research of Text Classification Based on TF-IDF and CNN-LSTM. *Journal of Physics: Conference Series*, *2171*(1), 012021. https://doi.org/10.1088/1742-6596/2171/1/012021

Zulfikar, W. B., Atmadja, A. R., & Pratama, S. F. (2023). Sentiment analysis on social media against public policy using multinomial naive bayes. *Scientific Journal of Informatics*, *10*(1), 25–34. https://doi.org/10.15294/SJI.V10I1.39952

IN PRESS

APPENDICES

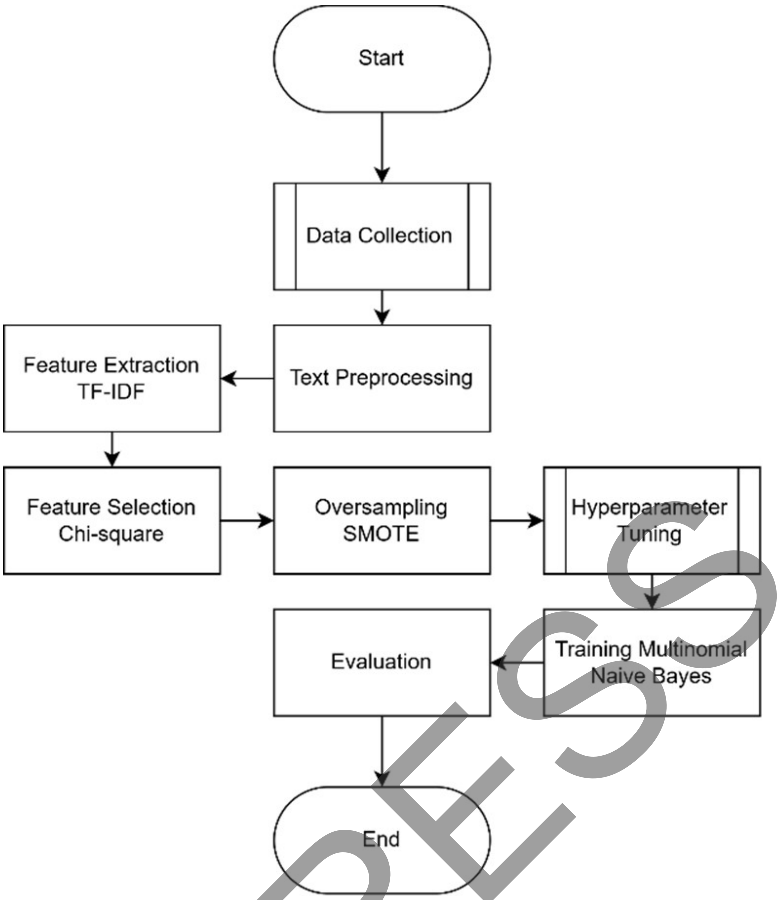


Figure 1 Research Flow

IN PRESS

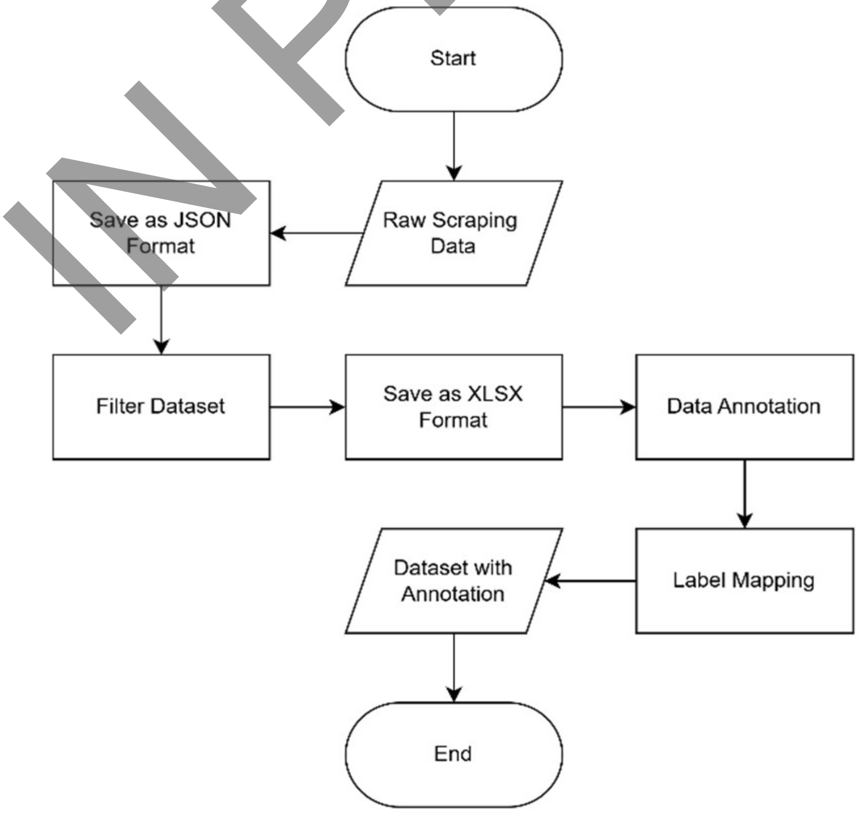


Figure 2 Data Collection Flow

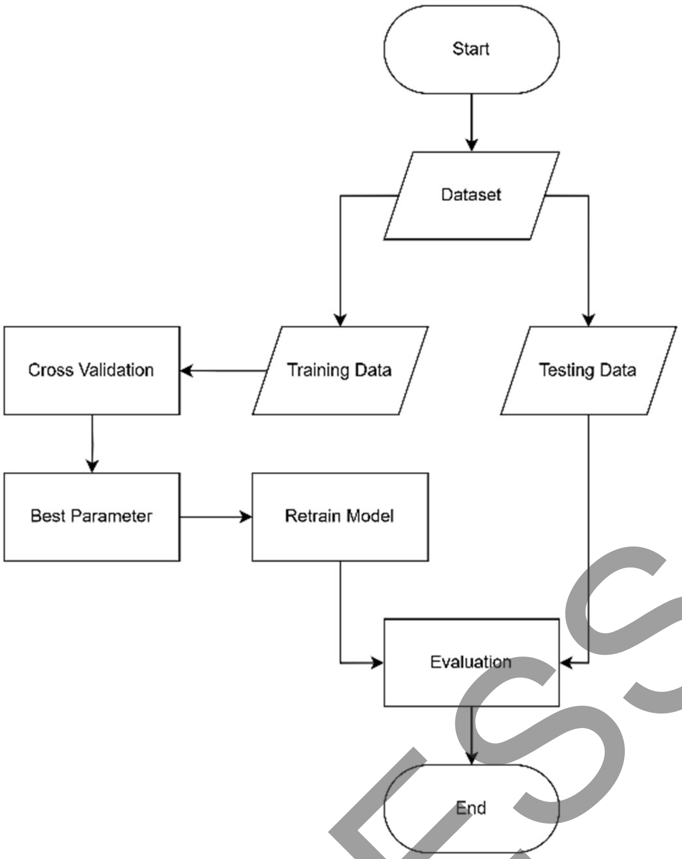


Figure 3 Hyperparameter Tuning Flow

IN PRESS

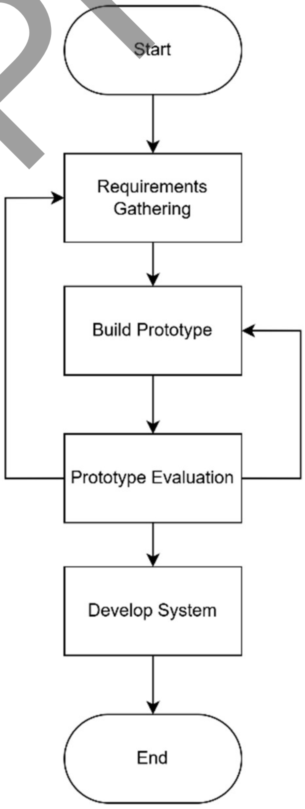
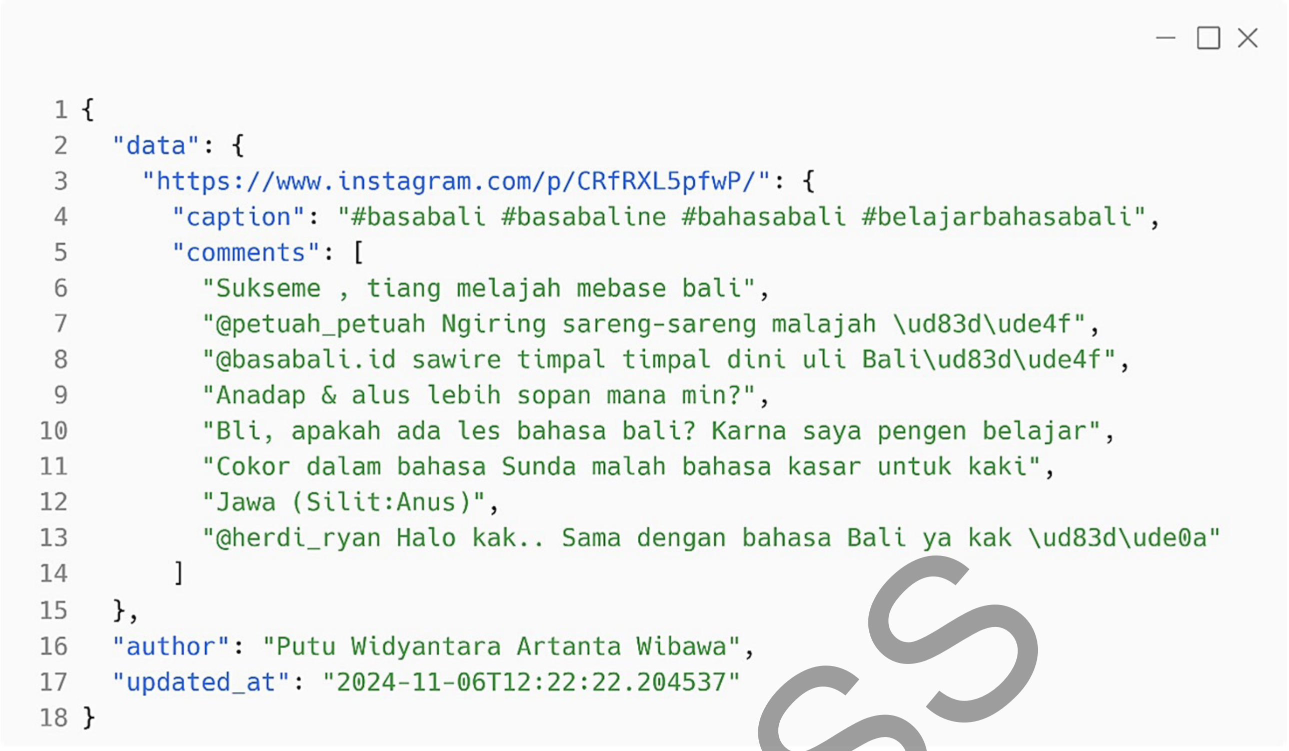


Figure 4 Prototyping Flow



```
{
  "data": {
    "https://www.instagram.com/p/CRfRXL5pfwP/": {
      "caption": "#basabali #basabaline #bahasabali #belajarbahasabali",
      "comments": [
        "Sukseme , tiang melajah mebase bali",
        "@petuah_petuah Ngiring sareng-sareng malajah \ud83d\ude4f",
        "@basabali.id sawire timpal timpal dini uli Bali\ud83d\ude4f",
        "Anadap & alus lebih sopan mana min?",
        "Bli, apakah ada les bahasa bali? Karna saya pengen belajar",
        "Cokor dalam bahasa Sunda malah bahasa kasar untuk kaki",
        "Jawa (Silit:Anus)",
        "@herdi_ryan Halo kak.. Sama dengan bahasa Bali ya kak \ud83d\ude0a"
      ]
    },
  "author": "Putu Widyantara Artanta Wibawa",
  "updated_at": "2024-11-06T12:22:22.204537"
}
```

Figure 5 Sample Dataset

IN PRESS

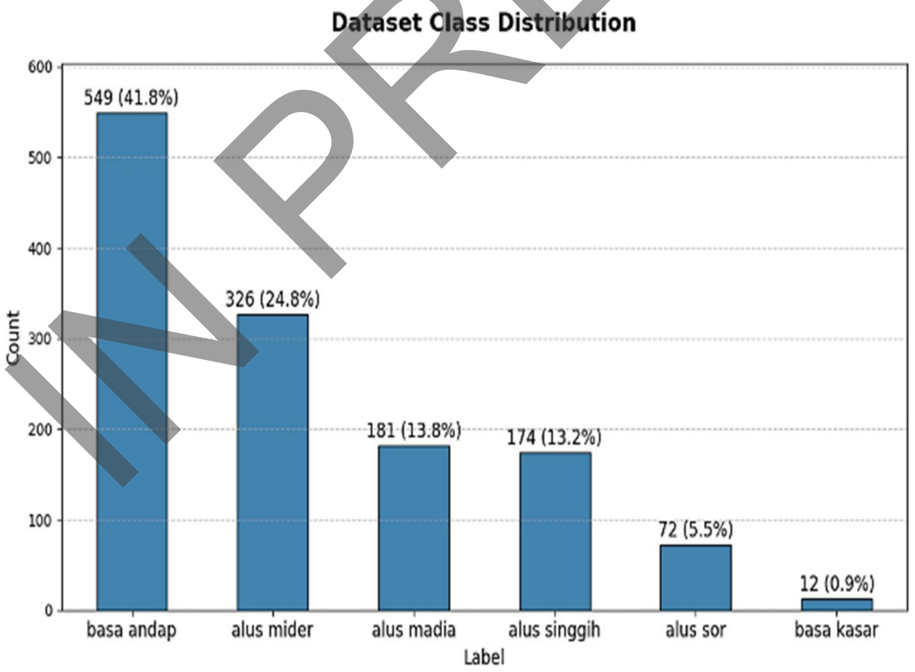


Figure 6 Dataset Class Distribution

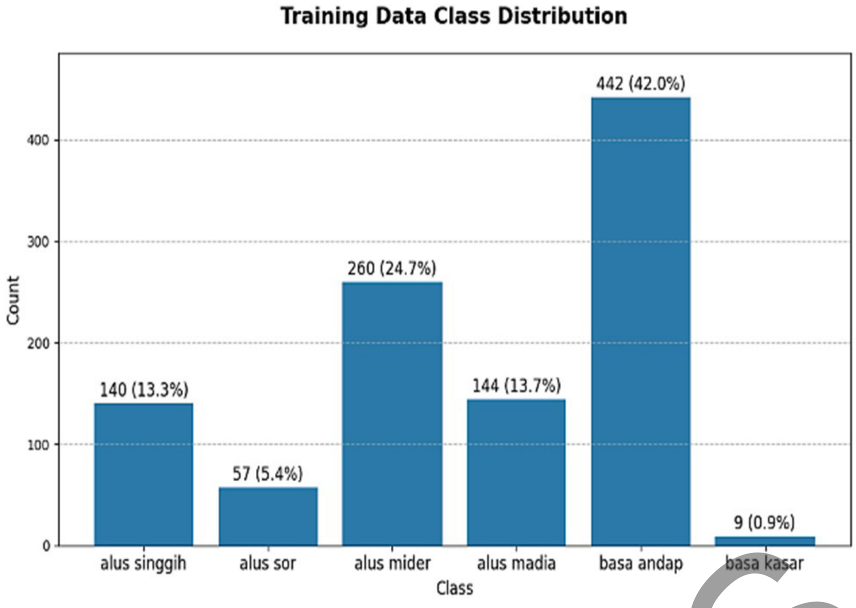
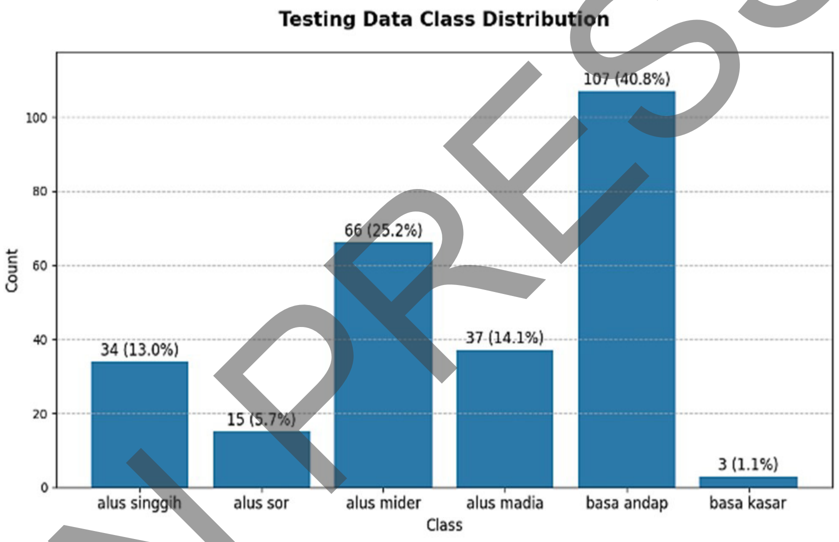


Figure 7 Training Data Class Distribution



IN PRESS

Figure 8 Testing Data Class Distribution

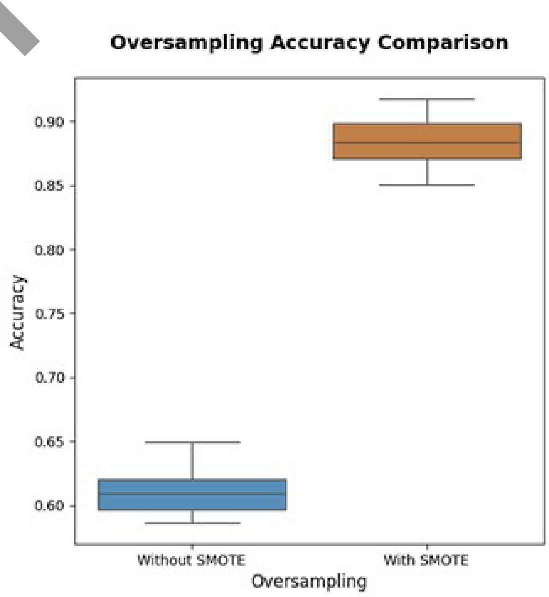


Figure 9 Oversampling Accuracy Comparison

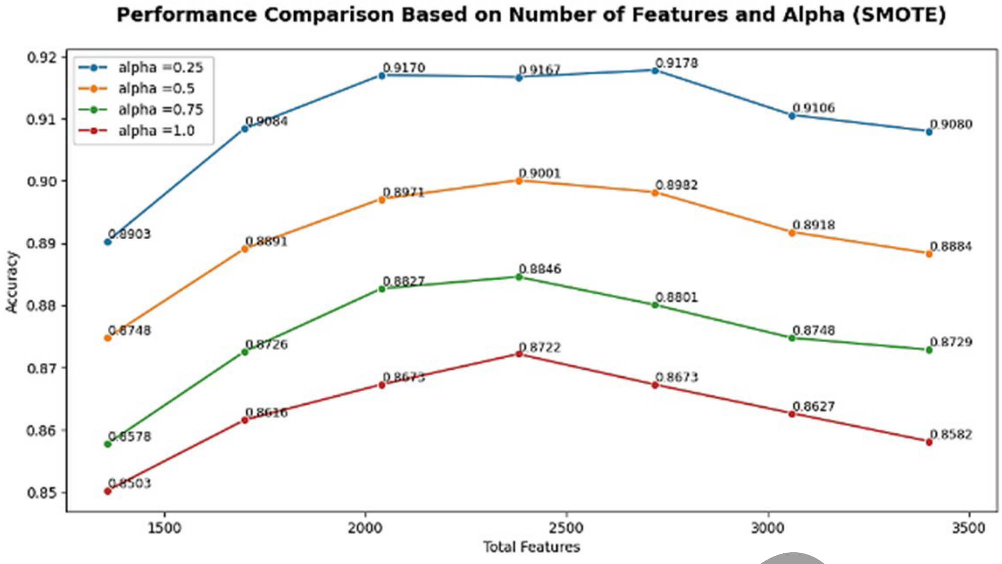


Figure 10 Performance Comparison Based on Number of Features and Alpha (SMOTE)

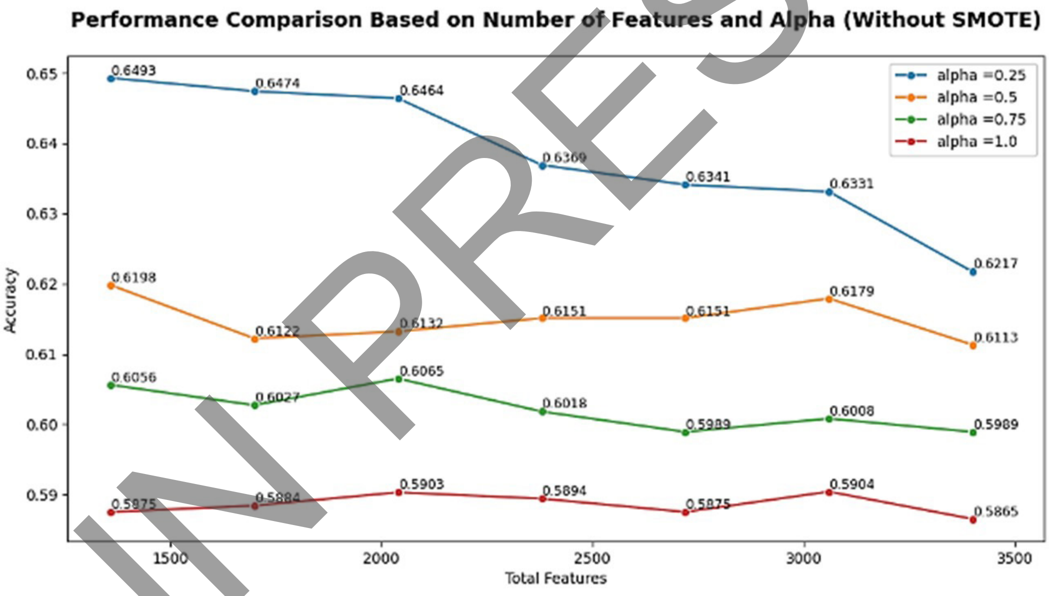


Figure 11 Performance Comparison Based on Number of Features and Alpha (Without SMOTE)

IN PRESS

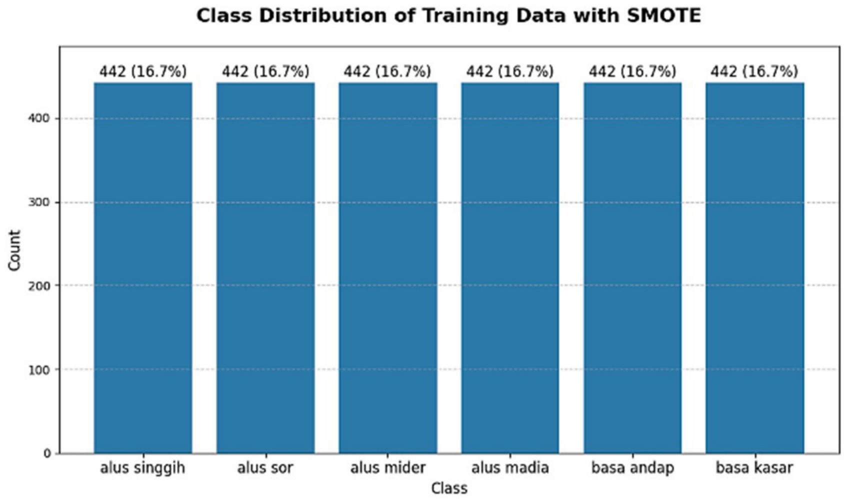


Figure 12 Class Distribution of Training Data with SMOTE

IN PRESS

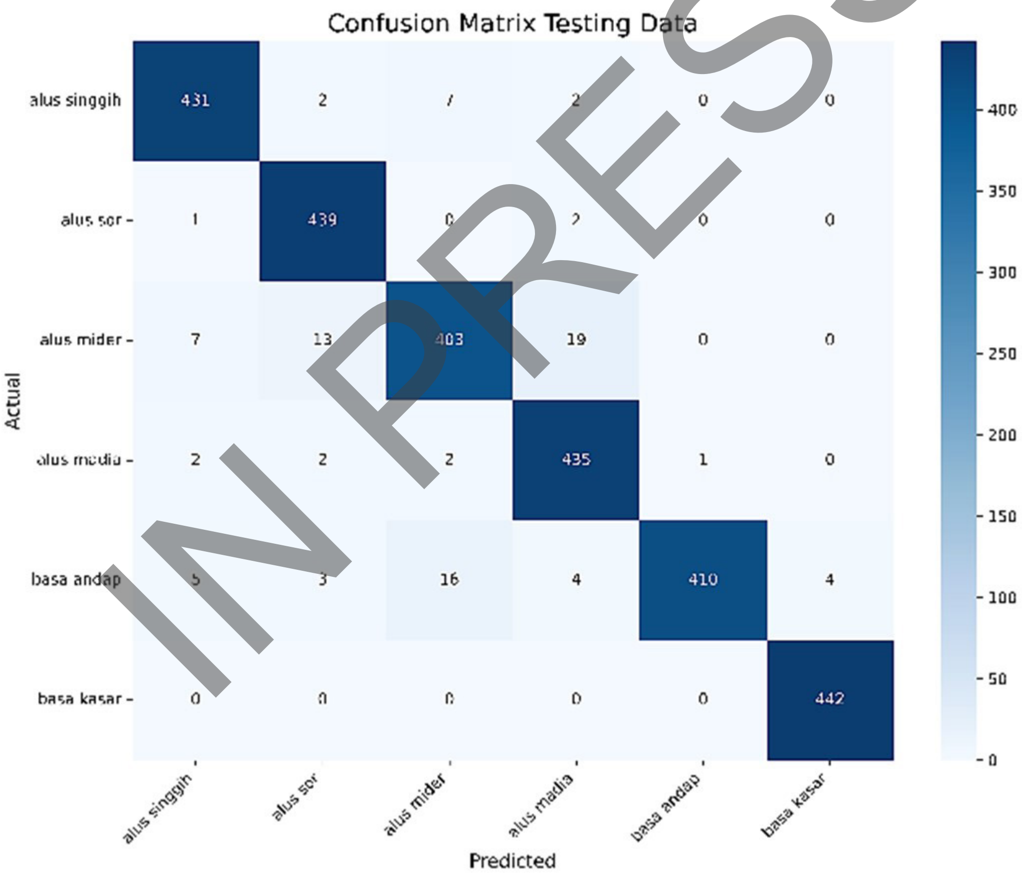


Figure 13 Confusion Matrix Training Data

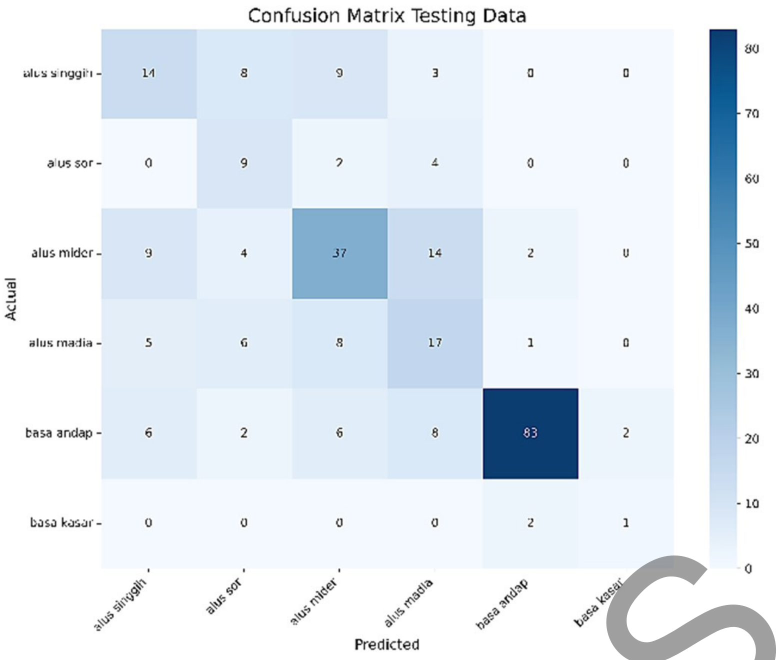
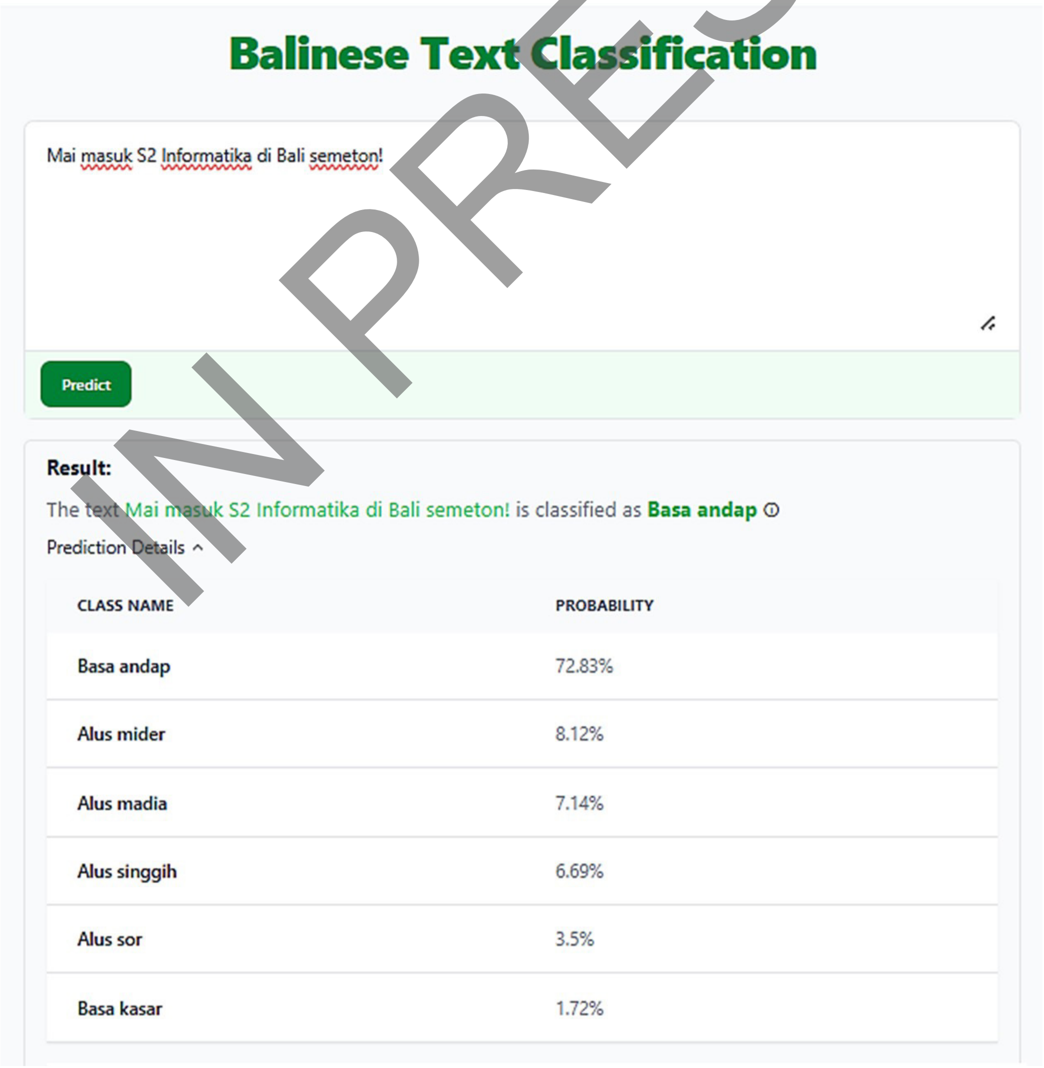


Figure 14 Confusion Matrix Testing Data



Balinese Text Classification

Mai masuk S2 Informatika di Bali semeton!

Result:

The text Mai masuk S2 Informatika di Bali semeton! is classified as **Basa andap** ⓘ

Prediction Details ^

CLASS NAME	PROBABILITY
Basa andap	72.83%
Alus mider	8.12%
Alus madia	7.14%
Alus singgih	6.69%
Alus sor	3.5%
Basa kasar	1.72%

Figure 15 System Interface

IN PRESS