Sentiment Analysis of Customer Review Using Classification Algorithms and SMOTE for Handling Imbalanced Class

N S Sediatmoko[1], Y Nataliani[*2], I Suryady[3]

[1,2]Department of Information System, Faculty of Information Technology, Satya Wacana Christian University, Salatiga, Indonesia

[3]Faculty of Information Technology, RMIT University, Melbourne, Australia

E-mail: nursiradjsediatmoko@gmail.com[1], yessica.nataliani@uksw.edu[2], isuryady@gmail.com[3]

Abstract. Ralali.com is a B2B e-commerce platform that offers various brands across categories ranging from automotive to building materials. The Play Store is a tool for downloading applications used by many people. This research aims to compare and find the best model among Naïve Bayes (NB), Support Vector Machine (SVM), and *k*-Nearest Neighbor (k-NN) in classifying the sentiment reviews of Ralali.com's application on the Play Store, and analyze the negative labels to provide recommendations for Ralali.com developers. The research methodology involves a classification approach, using these algorithms to handle the sentiment analysis task. Additionally, SMOTE (Synthetic Minority Over-sampling Technique) is applied to address class imbalance. Based on the research results, the NB Algorithm is the best choice compared to SVM and k-NN in addressing class imbalance. Using SMOTE generally improves the model performance on minority classes for Precision, Recall, and F-measure. However, there are some challenges to decrease the accuracy of non-SMOTE.

Keywords: sentiment analysis; classification; Naïve Bayes; support vector machine, k-Nearest Neighbor; SMOTE.

1. Introduction

In the era of globalization and technological advancement, e-commerce has become a primary pillar in transforming the way business is conducted. This phenomenon not only reflects technological development but also creates a new paradigm in trade and consumer interaction. E-commerce provides a platform that enables customers to explore and transact with products or services online, eliminating geographical barriers and expanding consumer accessibility to a variety of products [1]. In Indonesia, there are various e-commerce platforms with different business or trade models. One such existing model is Business-to-Business (B2B). B2B serves as a trading platform between companies to enhance effectiveness and efficiency in commercial relationships, whether in the supply chain, organizations, or industries [2]. One of the B2B e-commerce platforms in Indonesia is Ralali.com, facilitating online business transactions between companies through Ralali Marketplace.

Figure 1. The Top 10 E-commerce Highest Number of Visitors in Indonesia

In the second quarter of 2022, Databoks recorded the top 10 e-commerce platforms with the highest number of visitors in Indonesia. Ralali.com secured the 6th position as the e-commerce platform with the highest number of visitors, establishing itself as the largest B2B e-commerce platform in Indonesia [3]. Through Ralali Marketplace, Ralali.com offers a diverse range of products across various categories, including automotive and transportation; beauty, sports, and fashion; building materials; computers and communication; health and medical; Horeca; machinery and industrial spare parts; and others. Ralali Marketplace serves as an intermediary between sellers and buyers. Understanding user experience, including user opinions, is crucial for the success of any platform [4]. Sentiment analysis or opinion mining proves to be an effective method to comprehend and respond, enabling the platform to continuously evolve in creating a better and satisfying shopping experience for all parties involved.

Sentiment analysis, also known as opinion mining, is a field of study that analyzes individual opinions, sentiments, assessments, attitudes, and emotions towards entities such as products, services, organizations, individuals, issues, events, topics, and other attributes [5]. To understand consumer opinions, Ralali.com needs to conduct surveys and polls. However, with the development of the internet, the company can easily access reviews and public discussions related to the Ralali Marketplace platform on application distribution platforms, such as the Play Store. Although this review information is valuable, managing opinions or reviews scattered in the comment or review columns on the Play Store becomes a challenging task due to the multitude of opinions. Therefore, automated sentiment analysis is needed to summarize and analyze opinions effectively and efficiently to obtain various opinions, experiences, and feedback for Ralali.com.

Sentiment analysis can be conducted using various classification methods, with Naïve Bayes (NB), Support Vector Machine (SVM), and *k*-Nearest Neighbor (*k*-NN) being commonly employed. The following outlines several studies related to these classification methods. Sanjay et al. conducted research utilizing the NB and SVM algorithms for sentiment analysis of Amazon product reviews. The study revealed an accuracy of 84% for SVM and 82.875% for NB, showcasing superiority over conventional

techniques. Experimental results further confirm SVM's ability to discern feedback from Amazon products with a higher accuracy rate [6].

Research conducted by Wasim and Hassan classified opinions on the launch of the iPhone using the SVM algorithm, resulting in an accuracy of 89.21%. The conclusion of this research is that data collection using scraping methods in sentiment analysis can be done quickly, and the pre-processing stage of comments proves to be effective in generating sentences essential for sentiment analysis [7]. Subsequent research, conducted by Auliya, et al., implemented the *k*-NN algorithm for sentiment analysis in online learning. From this research, it was found that 56.24% of tweets had positive sentiment, while tweets with negative sentiment were 43.76%, with a total of 1825 data. This dataset was divided into two parts, namely 80% for training data and 20% for testing data, resulting in an accuracy rate of 84.93% in testing with K = 10. Auliya, et al., hope for further research to increase the amount of training data, implement larger methods, and use two or more methods to improve system accuracy [8].

In a study conducted by Pribadi et al., sentiment analysis of the PeduliLindungi application on Google Play revealed a negative trend. This indicates that during the use of the PeduliLindungi application, several issues still exist. Based on the results of the Random Forest algorithm testing, the implementation of Random Forest and SMOTE achieved an accuracy of 71%, recall of 70%, and precision of 70%. On the other hand, the implementation of Random Forest without SMOTE resulted in an accuracy of 60%, recall of 57%, and precision of 55%. Consequently, the implementation of SMOTE can enhance accuracy by 11%, recall by 13%, and precision by 15% [9].

In a study by Jakob et al., SMOTE consistently enhanced model performance, especially in managing high-class imbalance. Using SMOTE significantly improved the performance of SVM and Random Forest algorithms, particularly as data imbalance increased. Although there was no consistent superiority between SMOTE and ADASYN, both methods enhanced overall model performance. While SMOTE generally outperformed unprocessed data, it is important to note that in some cases, SMOTE could yield similar or worse results. Therefore, SMOTE is an effective tool for improving model performance on imbalanced datasets [10]. Failure to address class imbalance in classification tasks can greatly diminish classifier accuracy and performance. Machine learning models often struggle to predict minority class samples due to the focus on majority class samples, leading to detrimental effects on classification, especially in complex tasks. Real-world applications often involve biased data distributions, necessitating hybrid approaches to tackle class imbalance. Addressing class imbalance remains a crucial and evolving area of research in machine learning [11].

Based on previous studies, this study compares the performance of NB, SVM, and *k*-NN algorithms in classifying sentiment on a user review dataset from the Ralali.com Play Store. SMOTE is employed to address the imbalance in the dataset and ensure that the constructed model exhibits balanced performance to effectively recognize sentiments across all classes. The research focuses on data cleaning and pre-processing through two stages to produce high-quality and reliable data for modeling. The study's outcomes include determining the algorithm with the best accuracy, recall, precision, and f-measure rates, as well as analyzing negative label reviews to provide recommendations to the developers of Ralali Marketplace, namely Ralali.com.

2. Research Method

The stages conducted in this research are illustrated in Figure 2. The research flowchart shows ten stages, namely Data Scraping, Labeling, Cleaning and Pre-Processing Level 1, Data Checking Level 1, Cleaning and Pre-Processing Level 2, Data Checking Level 2, SMOTE, Split Data & Extract Features, Modeling, and Evaluation. The explanations for each stage are as follows:

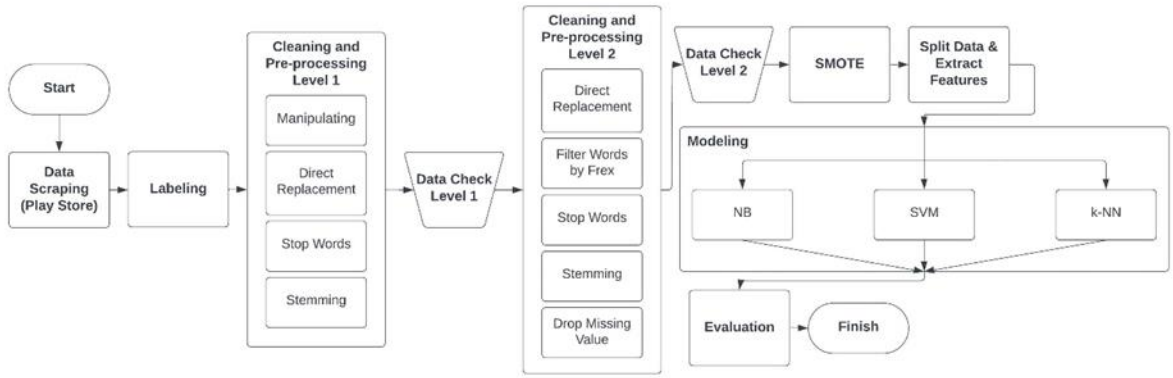

Figure 2. Research Flowchart

2.1. Data Scraping

Data scraping or web scraping is an automated data extraction technique from a website. This data can be utilized for sentiment analysis or other purposes [12]. In this research, the data retrieval process is conducted through the Play Store. The research dataset consists of text comments from the Play Store related to the Ralali Marketplace application. The scraping technique is implemented using the google_play_scraper library. There are a total of 11 attributes successfully scraped, i.e., reviewId, username, userImage, content, score, thumbsUpContent, reviewCreatedVersion, at, replyContent, repliedAt, and appVersion, but only the "content" and "score" attributes are utilized in this study. The "content" attribute contains information regarding user reviews, while the "score" attribute represents the application rating used for labeling purposes. Table 1 shows the information on Ralali data scraping.

Table 1. Information of Ralali Data Scraping

URL Application	Lang	Country	Scraping Date	Attributes	Range Index
com.ralali	Id	Id	January 8, 2024	reviewId, username, userImage, content, score, thumbsUpContent, reviewCreatedVersion, at, replyContent, repliedAt, appVersion	2206 entries

2.2. Labelling

Labelling is the process of assigning labels to characters used to identify a variable [13]. The labeling process consists of three classes: positive, negative, and neutral. Rating 5 and 4 is placed in the positive class, rating 3 in the neutral class, and ratings 1 and 2 in the negative class. The purpose of labeling is to divide the dataset into two parts, namely training data and testing data. Training data is utilized to train the system to recognize the sought patterns, while testing data is used to evaluate the results of the conducted training [7]. In Table 2, there is a text with each labeled as positive, negative, and neutral.

Table 2. Labelling Process

Text	Score	Label
Apliaksi platform yg memudahkan untuk bertransaksi di era digital.	5	Positive
Saya kesal karena kecamatan saya tidak terdaftar Di aplikasi ny padahal semuanya sudah benar!! Terus bagia verifikasi nya susah banget padahal udh di pencet tapi ttp aja gak bisa terkirim!!!!	3	Neutral

Text	Score	Label
Sudah lelah begadang untuk mendapatkan kuota promo tagihan Indihome, dan sudah terkonfirmasi via aplikasi akan dapat Cashback 20% max. 100rb. Ternyata setelah transaksi selesai Cashback tidak masuk. Chat ke CS diminta bukti SS Cashback yg terkonfirmasi. Sudah diberikan SS dr Aplikasi, CS minta yg di email. KALIAN SALAHKAN USER, BISA JADI SISTEM KALIAN YG LEMAH. TERBUKTI DARI REVIEW BINTANG SATU TENTANG LEMAHNYA RALALI. ⭐ Sy sudah lama pakai Ralali, tapi kali ini saya KECEWA SEKALI..!!	1	Negative

2.3. Cleaning and Pre-Processing Level 1

Pre-Processing is the initial step in machine learning where data is transformed or encoded to be brought into a state that allows machines to quickly navigate and parse the data [14]. The main objective of pre-processing is to detect, identify, and eliminate unwanted elements to enhance data quality for optimal performance [15]. In the Pre-Processing Level 1 stage, a series of steps are undertaken, including:

2.3.1. Manipulating

Data manipulation is performed to transform data based on several criteria, including converting characters to lowercase, removing non-ASCII characters, eliminating URLs, excluding mentions, separating strings based on capital letters, removing symbols, eradicating numbers, correcting duplications of three or more consecutive characters, eliminating double spaces, trimming leading and trailing spaces in sentences, as well as removing consecutive words of one or more [16].

2.3.2. Data Replacement

The purpose of this step is to replace inappropriate or non-standard words with correct words according to the KBBI rules [17]. This replacement is carried out to refine and clean the text, making it more comprehensible and enhancing the quality of the analysis or data processing involving the text. In Table 3, it shows the direct replacement process before and after.

Table 3. Direct Replacement

Before Words Replacement	After Words Replacement
aplilasi, aplikasih	aplikasi
pake, make	pakai
dwonload	download
g, ga, gak, gx, gax, nggak, ngak, tdk, ngga, tak, ngk, engga, kaga, tida, enggk	tidak
blom, belom, blm	belum

2.3.3. Stop Words

The removal of stop words is performed to cleanse the text from commonly used words that contribute minimally to the text analysis. Implementing stop words removal in the pre-processing stage can enhance the accuracy and performance of sentiment analysis [18].

2.3.4. Stemming

The final step at the pre-processing level 1 is stemming. In this process, all words are transformed into their base form by removing all affixes, including prefixes, suffixes, infixes, and combinations of prefix and suffix or circumfixes [8]. In Table 4, it shows the cleaning and pre-processing process before and after. However, the stemming process for words originating from Indonesian, especially slang or informal words

originating from foreign languages such as Javanese, irregular words, and acronyms, has some limitations. In this study, the word "fiturnya" does not change to "fitur". This may be considered a complete form and not changed because the affix "nya" is part of the word in a specific context [19].

Table 4. Cleaning and Pre-Processing Level 1

Content	Cleaning and Pre-Processing Level 1
Aplikasinya keren si, fiturnya mudah di fahami, no ribet2, apalagi usernya friendly2 jadi makin nyaman, Barangnya juga murah2 dan banyak promonya, dan yang paling penting pengirimannya cepet, pokoknya mantep betul, sya kasi bintang 5,,	aplikasi keren si fiturnya mudah paham nomor ribet usernya friendly jadi makin nyaman barang murah banyak promonya yang paling penting kirim cepat pokok mantap betul kasi bintang
Aplikasi ini memiliki antarmuka pengguna yang intuitif dan ramah pengguna, dengan tampilan yang bersih dan desain responsif. Fitur pencariannya sangat efektif, memungkinkan saya dengan mudah menemukan produk yang saya cari. Selain itu, kemudahan bertransaksi dan berbagai metode pembayaran yang tersedia membuat pengalaman berbelanja menjadi lebih nyaman dan aman. bagi yang belum instal karena sangat bagus 👌	aplikasi milik antarmuka guna intuitif ramah guna tampil bersih desain responsif fitur cari sangat efektif mungkin dengan mudah temu produk saya cari itu mudah transaksi bagai metode bayar sedia buat alam belanja jadi lebih nyaman aman yang install sangat bagus
Jangan transaksi apa pun pakai aplikasi ini.. Karena transaksi anda tidak akan di proses dan dana anda tidak di kembalikan... Padahal sudah berhasil di transfer.. Tidak ada pertanggungjawaban uang kembali atas setiap transaksi, ada pengaduan pun percuma online, tapi tidak di tanggapi...	jangan transaksi apa pakai aplikasi karena transaksi proses dana tidak kembali padahal hasil transfer tidak pertanggungjawaban uang atas transaksi ada adu percuma online tidak di tanggap

2.4. Data Checking Level 1

In the process of cleaning and pre-processing data, a crucial step taken is meticulous data checking. Data checking is conducted to ensure that each data element or text aligns with the expectations and requirements of the upcoming analysis [20]. The main objective of this stage is to ensure the accuracy and integrity of the data to be used in the research. Through thorough data checking, we can minimize the potential for errors or inconsistencies, resulting in a clean, accurate dataset ready for further analysis.

2.5. Cleaning and Pre-Processing Level 2

In analyzing the sentiment of text comments on the Play Store, data cleaning and pre-processing are conducted twice. The first step involves handling heterogeneity, consistency in format, and text structure by replacing words not in accordance with KBBI, removing stop words, and performing stemming. The second step focuses on addressing noise such as slang and abbreviations to enhance sentiment analysis algorithm accuracy, by removing rarely occurring words, performing stop words and stemming again, and deleting rows containing empty text. This approach ensures a more focused dataset, improves sentiment interpretation without interference or bias, and optimizes the results of sentiment analysis. In Table 5, the cleaning and pre-processing level 2 process is illustrated before and after the level 1 process.

Table 5. Cleaning and Pre-Processing Level 2

Cleaning and Pre-Processing Level 1	Cleaning and Pre-Processing Level 2
buruk sekali pelayananya hubung customer service via wa hari balas keren promo nya	buruk sekali layan hubung customer service via whatsapp hari balas keren promo

2.6. Data Checking Level 2

Ensuring the accuracy and integrity of the data is crucial to produce a clean, accurate dataset ready for further analysis.

2.7. SMOTE

The application of the Synthetic Minority Over-sampling Technique (SMOTE) in this research is imperative due to the imbalanced distribution in sentiment analysis. SMOTE is utilized to balance the class distribution, thereby significantly improving the performance of the classification model [21]. This technique contributes positively to the accuracy and reliability of the model in identifying sentiments in an imbalanced dataset. By employing SMOTE, the achieved balance ensures that the interpretation of sentiment analysis results is more representative and reliable to support the findings of this research.

2.8. Split Data & Extract Features

The dataset is divided into two parts, namely training and testing, with a weight of 80% for training and 20% for testing. The purpose of this data split is to address overfitting, allowing the evaluation of algorithm performance during the testing phase [21]. In supervised learning, dividing the dataset into training data plays a crucial role in executing the learning phase, which is then used as a reference for predictions by the algorithm [7]. Feature extraction is a technique in data management to reduce its complexity, and this technique is utilized to extract the most important information from high-dimensional data. Feature selection can significantly contribute to the performance of machine learning models [22]. The feature extraction method applied in this research employs Bag of Words (BoW) using 'CountVectorizer'. The BoW method represents each word in the text by counting and converting it into a numeric vector. This technique has proven to be a robust approach in sentiment analysis [23].

2.9. Modeling

In this research, after separating the data and extracting features, the next step is to perform classification using three algorithms: NB, SVM, and *k*-NN. We used the MultinomialNB model for NB, fitting it with the resampled training data (X_train_resampled, y_train_resampled) and predicting the test data (X_test_vectorized). The model's performance was evaluated using a classification report. For SVM, we used a linear kernel and trained the model similarly on the resampled training data, then predicted the test data, with the performance assessed through a classification report. For *k*-NN, we defined a parameter grid for the number of neighbors (n_neighbors) and performed a grid search with 5-fold cross-validation to find the best parameters. The best model was then used to predict the test data, and the classification report was generated to evaluate its performance. All implementations were done using Scikit-learn in Python, ensuring consistent preprocessing and parameter tuning to optimize performance. The performance of these three algorithms was compared and evaluated by measuring accuracy, recall, precision, and the F-measure at the evaluation stage.

2.10. Evaluating

The final stage involves evaluation, where the three algorithms are compared based on accuracy, recall, precision, and the f-measure. Additionally, the negative class will be further analyzed to provide recommendations to the developers of Ralali Marketplace, namely Ralali.com.

3. Result

After performing data cleaning and pre-processing, resulting in 1964 data points, the next step in sentiment analysis is to divide the data into training data (80%) and testing data (20%). After the data splitting process is completed, the dataset's features are extracted using Count Vectorizer. Figure 3 illustrates the data splitting and feature extraction process.

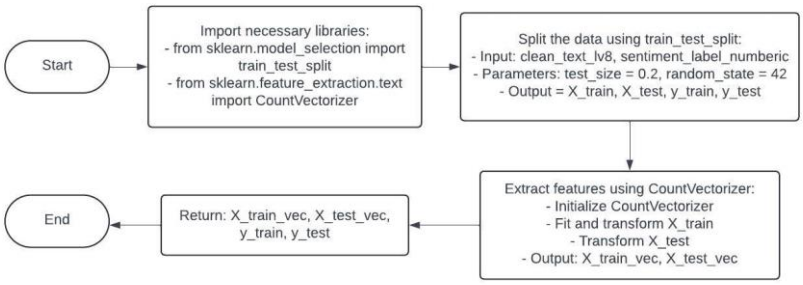

Figure 3. Split Data and Extract Features Algorithms

After the data splitting and feature extraction, the SMOTE method is applied to address class imbalance. From 2964 data points, there are 1504 data points in the positive class, 391 data points in the negative class, and 69 data points in the neutral class. By using `sampling_strategy='auto'`, SMOTE adjusts the number of synthetic samples generated for the minority class to balance with the majority class. The oversampling results are stored in `X_train_resampled` and `y_train_resampled`, which can then be used to train a machine learning model to significantly improve the classification model's performance. Figure 4 illustrates the process of using SMOTE.

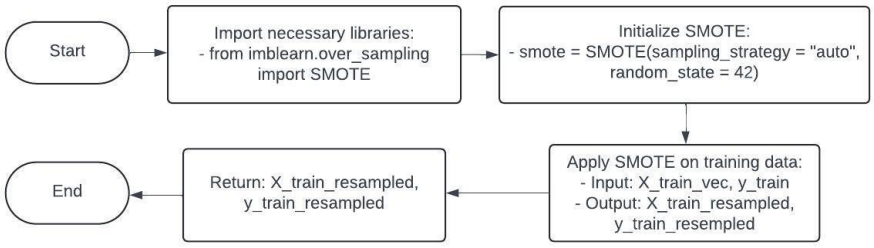

Figure 4. SMOTE Process

All calculations in this research were conducted using the Python programming language with the Jupyter Notebook interface and Anaconda Navigator. Calculation formulas or libraries are accessible as open source and can be used for developers' purposes. Following the SMOTE process, modeling was performed using the NB, SVM, and *k*-NN algorithms. These three algorithms will be compared through calculations of accuracy, recall, precision, and f-measure.

In Table 6, a confusion matrix is used to evaluate the performance of the classification model in predicting the sentiment of a text. This matrix includes three sentiment classes: positive, neutral, and negative. True Positive (TP) indicates the number of texts correctly predicted as positive by the model. False Positive (FP) refers to the number of texts incorrectly predicted as positive by the model. True Negative (TN) is the number of texts correctly predicted as not positive by the model. Meanwhile, False Negative (FN) is the number of texts that should have been predicted as positive but were incorrectly predicted by the model. By using these values, we can calculate evaluation metrics such as accuracy, precision, recall, and f-Measure for each sentiment class.

Table 6. Confusion Matrix

Actual/Prediction	Positive	Neutral	Negative
Positive	TP	FN	FP
Neutral	FP	TN	FN
Negative	FN	FP	TN

3.1.1. Accuracy

The accuracy in a classification model is the level of agreement between the predicted results and the actual values [24]. The confusion matrix from Table 6 is used to calculate the percentage accuracy of the classification model with Eq. (1).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \quad (1)$$

Table 7 shows the accuracy results of three different classification algorithms, namely NB, SVM, and *k*-NN, both with and without using the SMOTE method. The experimental results indicate that for NB, the accuracy is higher without using SMOTE (87%) compared to using SMOTE (85%). On the other hand, for SVM, the accuracy is lower when using SMOTE (83%) compared to without SMOTE (89%). For *k*-NN, using SMOTE results in lower accuracy (60%) compared to without SMOTE (82%). These results indicate that the use of SMOTE can affect the performance of classification algorithms differently depending on the algorithm used.

Table 7. Result of Accuracy Each Algorithm

Algorithm	NB	SVM	*k*-NN
Non-SMOTE	87%	89%	82%
SMOTE	85%	83%	60%

3.1.2. Precision

Precision in evaluating the performance of machine learning not only provides information about the errors made by the algorithm but also describes the model's ability to correctly identify positive examples out of all examples predicted as positive [25]. Precision is calculated as the result of the number of true positive predictions (TP) divided by the total positive predictions made by the system. The formula for precision is given in Eq. (2).

$$Precision = \frac{TP}{TP+FP} \times 100\% \quad (2)$$

Table 8 presents the precision results of three different classification algorithms, namely NB, SVM, and *k*-NN, implemented with and without using the SMOTE method. The experimental results indicate that for NB, the precision is higher with SMOTE (62%) than without SMOTE (54%). For SVM, the precision is also higher with SMOTE (60%) than without SMOTE (56%). However, for *k*-NN, the precision is higher when using without SMOTE (52%) than without SMOTE (46%). This indicates that the use of the SMOTE method can have a positive or negative impact depending on the classification algorithm used.

Table 8. Result of Precision Each Algorithm

Algorithm	NB	SVM	*k*-NN
Non-SMOTE	54%	56%	52%
SMOTE	62%	60%	46%

3.1.3. Recall

Recall is calculated as the division between the number of correct data and the total number of expected data [26]. The recall formula can be seen in Eq. (3).

$$Recall = \frac{TP}{TP+TN} \times 100\% \quad (3)$$

Table 9 presents the recall results of three different classification algorithms, namely NB, SVM, and *k*-NN, run with and without using the SMOTE method. The experimental results indicate that for NB, the recall is higher with SMOTE (69%) compared to without SMOTE (59%). For SVM, the recall is also higher with SMOTE (67%) compared to without SMOTE (57%). Meanwhile, for *k*-NN, the recall is higher when using SMOTE (54%) compared to without SMOTE (47%). These findings suggest that the use of the SMOTE method can improve recall in some classification algorithms, but its impact may vary depending on the type of algorithm used.

Table 9. Result of Recall Each Algorithm

Algorithm	NB	SVM	*k*-NN
Non-SMOTE	59%	57%	47%
SMOTE	69%	67%	54%

3.1.4. F-Measure

The F-Measure, often referred to as the harmonic mean, combines precision and recall to reflect the overall performance of a classification model [27]. The F-1 formula is given in Eq. (4).

$$F - Measure = \frac{2 \times Precision \times Recall}{Recall + Precision} \quad (4)$$

Table 10 shows the F-Measure values of three different classification algorithms, namely NB, SVM, and *k*-NN, run with and without using the SMOTE method. The F-Measure combines precision and recall to provide an overview of the overall performance of a classification model. The experimental results indicate that for NB, the F-Measure is higher with SMOTE (64%) compared to without SMOTE (57%). For SVM, the F-Measure is also higher with SMOTE (62%) compared to without SMOTE (56%). However, for *k*-NN, the F-Measure is lower when using SMOTE (47%) compared to without SMOTE (48%). These findings suggest that the use of the SMOTE method can impact the F-Measure of classification algorithms, with varying effects depending on the algorithm used.

Table 10. Result of F-Measure Each Algorithm

Algorithm	NB	SVM	*k*-NN
Non-SMOTE	57%	56%	58%
SMOTE	64%	62%	47%

4. Discussion

The analysis of Figure 5 shows that the performance of various types of models exhibits a significant improvement on datasets using the SMOTE compared to without SMOTE datasets. This improvement is particularly evident in the metrics of precision, recall, and f-measure. However, there is a decrease in the accuracy metric for all algorithms when using the SMOTE technique. These results indicate that the consistent use of SMOTE can enhance the model's ability to classify minority data, albeit with a slight sacrifice in overall accuracy. NB algorithm shows superior performance compared to other algorithms after

applying the SMOTE technique. This indicates that NB is a good choice for addressing class imbalance in the dataset used in this study.

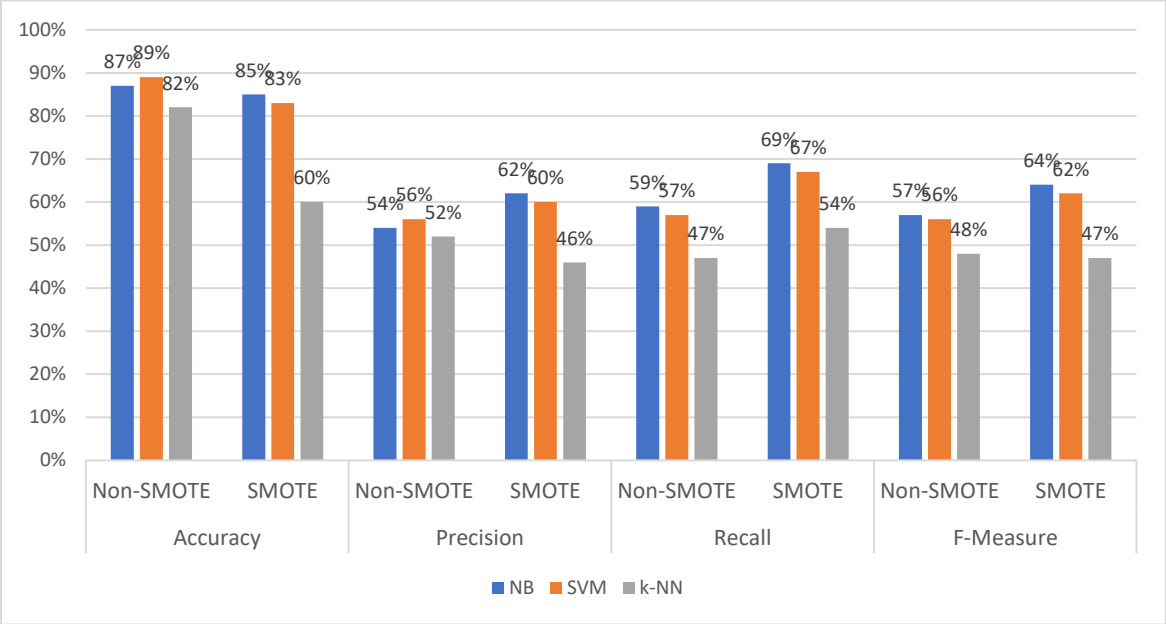

Figure 5. Comparison of Classification Model Performance with and without SMOTE Utilization on Evaluation Metrics: Accuracy, Precision, Recall, and F-Measure

The analysis in Table 11 reveals a significant difference between the use of SMOTE and Non-SMOTE techniques. The results indicate that the use of SMOTE positively impacts the model's performance on minority or neutral classes, as reflected in the increased precision, recall, and f-measure compared to without SMOTE models. However, the precision values are still relatively low for all classes. These findings provide empirical support for the effectiveness of the SMOTE technique in enhancing model performance on minority classes, consistent with previous research by Mabunda et al. [21]. In the study conducted by Javad et al., it is important to note that while SMOTE is a common and robust technique for balancing datasets with an extremely unbalanced ratio, it may only sometimes increase accuracy. In fact, SMOTE can sometimes decrease accuracy, as noted in the study. This highlights the need for careful consideration and evaluation of the impact of SMOTE on model performance, especially in the context of imbalanced datasets [28].

Table 11. Precision, Recall, and F-Measure for Imbalance Neutral Class

	Precision		**Recall**		**F-Measure**	
	Non-SMOTE	**SMOTE**	**Non-SMOTE**	**SMOTE**	**Non-SMOTE**	**SMOTE**
NB	0%	22%	0%	44%	0%	29%
SVM	0%	23%	0%	38%	0%	29%
***k*-NN**	0%	8%	0%	38%	0%	13%

The analysis of Figure 6 shows the model's performance in predicting three sentiment classes, as revealed by the confusion matrices: class 0 (negative), class 1 (neutral), and class 2 (positive). In the non-SMOTE models, the positive class shows high accuracy, especially in the NB model at 93%, but the negative and neutral classes suffer from significant misclassifications. The SVM model achieves 96% accuracy for the positive class and 74% for the negative class but only 0% for the neutral class. The *k*-NN

model has 96% accuracy for the positive class, with the negative class accuracy dropping to 41% and only 0% for the neutral class. After applying SMOTE, the SMOTE-NB model shows balanced improvements with 89% accuracy for the positive class, 75% for the negative class, and 44% for the neutral class. The SMOTE-SVM model remains similar, with 87% accuracy for the positive class, 75% for the negative class, and 38% for the neutral class. The SMOTE-*k*NN model achieves more balanced results, with 61% accuracy for the positive class, 64% for the negative class, and 38% for the neutral class. These results indicate that while SMOTE improves class balance, further refinement is needed to enhance accuracy and reduce misclassifications, particularly for neutral sentiments.

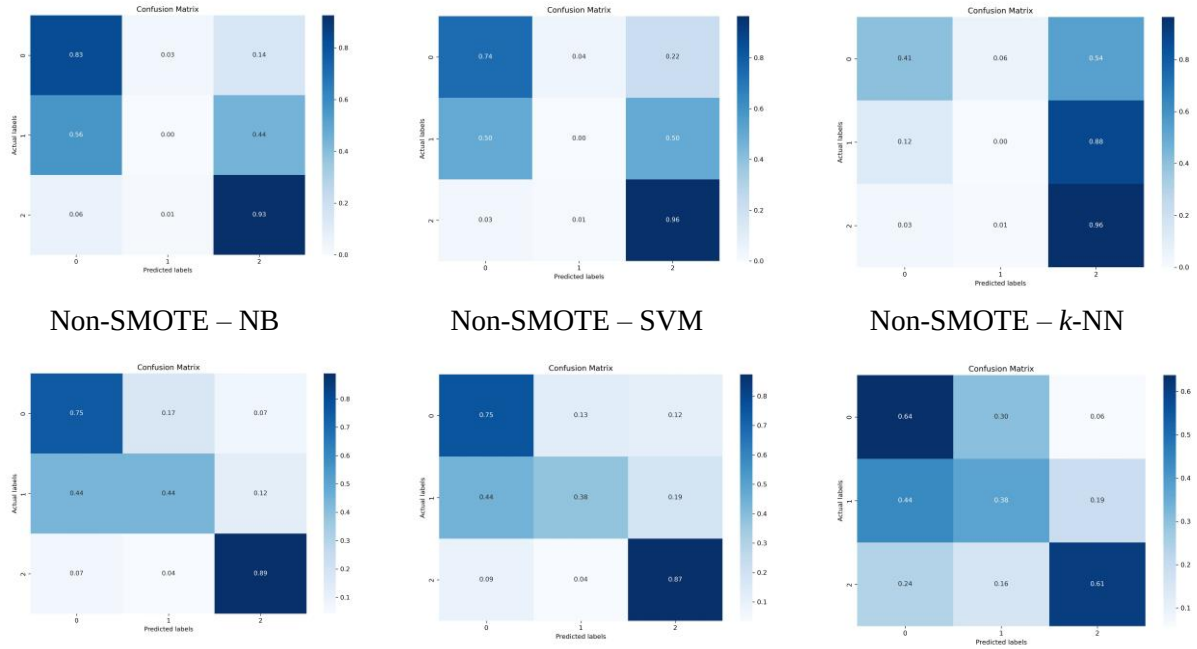

Figure 6. Confusion Matrix for NB, SVM, and *k*-NN

In sentiment analysis of comments on the Ralali Marketplace app in the Play Store, this study classified sentiment into three classes: positive, negative, and neutral. This classification is based on the ratings given, where ratings 4 and 5 are classified as positive, rating 3 as neutral, and ratings 1 and 2 as negative. The current focus is on the negative class, with the aim of providing suggestions to the developers of the Ralali Marketplace for improvement. Word Cloud analysis of the negative class shows frequently occurring words, including: application, transaction, payment, login, money, order, goods, long, and refund. Figure 7 shows the Word Cloud in the sentiment analysis of the Ralali Marketplace app.

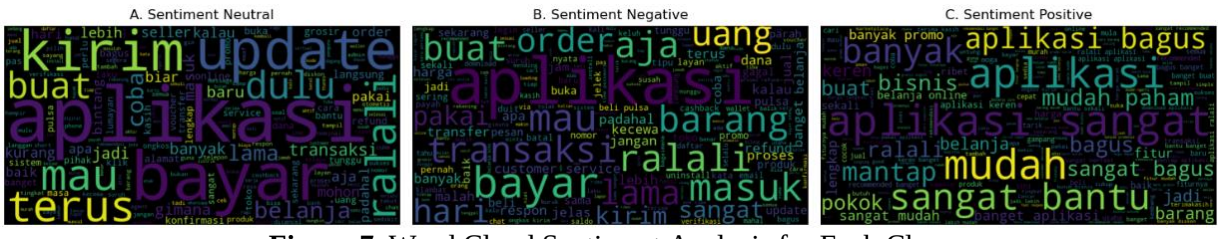

Figure 7. Word Cloud Sentiment Analysis for Each Class

In conducting sentiment analysis on user comments and ratings, especially those classified as negative (ratings 1 and 2), several feedback points were found that could be considered for the development of the Ralali Marketplace application. First, users complained about the slow and opaque refund process. Developers should improve the refund process to be faster and easier, even considering automatic refund services to enhance user satisfaction. Second, some users encountered difficulties during registration and account verification. Some complained that after registration, the application kept loading and could not be used. Developers should improve the registration and verification process to be smoother and faster, and optimize the application performance to avoid ongoing loading issues. Third, some users experienced difficulties in making payments, especially in paying bills such as PDAM. Some users also suggested simplifying the menu layout to be more intuitive and user-friendly. Developers should improve the payment system and simplify the menu layout so that users can transact more easily and smoothly.

The basis for the recommendations provided to developers stems from the results and insights gained from the analysis and evaluation of the classification models used in this research. Specifically, the recommendations are based on the performance metrics analysis, including accuracy, precision, recall, and f-measure for NB, SVM, and *k*-NN algorithms. The NB algorithm, combined with SMOTE, achieved the highest balance of these metrics, indicating its effectiveness in handling imbalanced datasets. This directly informs the recommendation to use NB with SMOTE for similar data scenarios. Additionally, the significant improvement in model performance due to the two-stage data cleaning and preprocessing process highlights the importance of these steps, thus underpinning the recommendation to ensure thorough data cleaning and preprocessing.

The challenges encountered in handling slang, informal words, words from foreign languages like Javanese, irregular words, and acronyms during the Word Cloud analysis underscore the need for tailored preprocessing steps. These insights lead to the recommendation for developers to incorporate specialized preprocessing techniques when dealing with such data types. Moreover, user feedback and experiences gathered through various feedback mechanisms and Word Cloud analysis provided additional context for recommendations aimed at improving the refund process, registration and verification process, payment system, and menu layout. These recommendations are also supported by established best practices and findings in related literature, ensuring that the suggested improvements are both innovative and grounded in proven methods. By integrating these insights and findings, the recommendations aim to enhance user experience, improve customer satisfaction, and optimize the platform's performance, ultimately benefiting the Ralali.com e-commerce platform.

5. Conclusion

The conclusion of this research indicates that using SMOTE generally improves the model performance on minority classes for precision, recall, and f-measure. However, there are still challenges related to lower precision compared to using non-SMOTE. Performing data cleaning and pre-processing in two stages is crucial as it significantly enhances the model's performance, effectively handles slang or informal words, words from foreign languages like Javanese, irregular words, and acronyms, and positively impacts providing feedback to developers when using Word Cloud. The NB algorithm stands out as the best choice compared to SVM and k-NN in addressing class imbalance, with SMOTE combined with the NB algorithm resulting in an accuracy of 85%, precision of 62%, recall of 69%, and f-measure of 64%. Based on these results, it is recommended that Ralali Marketplace developers improve the refund process to be faster and easier, enhance the registration and verification process to be smoother and faster, improve the payment system, and simplify the menu layout for users to transact more efficiently and smoothly. By understanding users' views and perceptions of the platform, application developers can identify issues, improve user experience, measure customer satisfaction, enhance reputation, increase user retention, and develop marketing strategies, which can benefit the Ralali.com e-commerce platform. The development prospects of this research include analyzing different training and testing data comparisons. Furthermore, future

research could consider other methods besides SMOTE to handle data imbalance and focus more on handling extremely imbalanced data or significant differences between majority and minority classes.

6. References

[1] P. Dolfen *et al.*, "Assessing the Gains from E-Commerce," *Am. Econ. J. Macroecon.*, vol. 15, no. 1, pp. 342–370, 2023, doi: 10.1257/mac.20210049.

[2] C. Loro and R. Mangiaracina, "The impact of e-marketplace on the B2b relationships," *Ind. Manag. Data Syst.*, vol. 122, no. 1, pp. 37–54, 2022, doi: 10.1108/IMDS-11-2020-0651.

[3] A. Ahdiat, "10 E-commerce dengan Pengunjung Terbanyak Kuartal II 2022." iPrice, 2022. [Online]. Available: https://iprice.co.id/insights/mapofecomm

[4] B. Yang, Y. Liu, Y. Liang, and M. Tang, "Exploiting user experience from online customer reviews for product design," *Int. J. Inf. Manage.*, vol. 46, no. May 2018, pp. 173–186, 2019, doi: 10.1016/j.ijinfomgt.2018.12.006.

[5] S. Bhatia, M. Sharma, and K. K. Bhatia, "Sentiment Analysis and Mining of Opinions," *Stud. Big Data*, vol. 30, no. May, pp. 503–523, 2018, doi: 10.1007/978-3-319-60435-0_20.

[6] S. Dey, S. Wasif, D. S. Tonmoy, S. Sultana, J. Sarkar, and M. Dey, "A Comparative Study of Support Vector Machine and Naive Bayes Classifier for Sentiment Analysis on Amazon Product Reviews," *2020 Int. Conf. Contemp. Comput. Appl. IC3A 2020*, pp. 217–220, 2020, doi: 10.1109/IC3A48958.2020.233300.

[7] W. Bourequat and H. Mourad, "Sentiment Analysis Approach for Analyzing iPhone Release using Support Vector Machine," *Int. J. Adv. Data Inf. Syst.*, vol. 2, no. 1, pp. 36–44, 2021, doi: 10.25008/ijadis.v2i1.1216.

[8] A. R. Isnain, J. Supriyanto, and M. P. Kharisma, "Implementation of K-Nearest Neighbor (K-NN) Algorithm For Public Sentiment Analysis of Online Learning," *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 15, no. 2, p. 121, 2021, doi: 10.22146/ijccs.65176.

[9] M. R. Pribadi, D. Manongga, H. D. Purnomo, I. Setyawan, and Hendry, "Sentiment Analysis of the PeduliLindungi on Google Play using the Random Forest Algorithm with SMOTE," *2022 Int. Semin. Intell. Technol. Its Appl. Adv. Innov. Electr. Syst. Humanit. ISITIA 2022 - Proceeding*, no. November, pp. 115–119, 2022, doi: 10.1109/ISITIA56226.2022.9855372.

[10] J. Brandt and E. Lanzén, "A Comparative Review of SMOTE and ADASYN in Imbalanced Data Classification," *2021*, p. 42, 2020, [Online]. Available: https://www.diva-portal.org/smash/record.jsf?pid=diva2:1519153

[11] P. Kumar, R. Bhatnagar, K. Gaur, and A. Bhatnagar, "Classification of Imbalanced Data:Review of Methods and Applications," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1099, no. 1, p. 012077, 2021, doi: 10.1088/1757-899x/1099/1/012077.

[12] M. Birjali, M. Kasri, and A. Beni-Hssane, "A comprehensive survey on sentiment analysis: Approaches, challenges and trends," *Knowledge-Based Syst.*, vol. 226, p. 107134, 2021, doi: 10.1016/j.knosys.2021.107134.

[13] Z. Rais, F. T. T. Hakiki, and R. Aprianti, "Sentiment Analysis of Peduli Lindungi Application Using the Naive Bayes Method," *SAINSMAT J. Appl. Sci. Math. Its Educ.*, vol. 11, no. 1, pp. 23–29, 2022, doi: 10.35877/sainsmat794.

[14] J. Hartmann, M. Heitmann, C. Siebert, and C. Schamp, "More than a Feeling: Accuracy and Application of Sentiment Analysis," *Int. J. Res. Mark.*, vol. 40, no. 1, pp. 75–87, 2023, doi: 10.1016/j.ijresmar.2022.05.005.

[15] A. M. S. Shaik Afzal, "Optimized support vector machine model for visual sentiment analysis," *2021 3rd Int. Conf. Signal Process. Commun. ICPSC 2021*, no. May, pp. 171–175, 2021, doi: 10.1109/ICSPC51351.2021.9451669.

[16] H. T. Duong and T. A. Nguyen-Thi, "A review: preprocessing techniques and data augmentation for sentiment analysis," *Comput. Soc. Networks*, vol. 8, no. 1, pp. 1–16, 2021, doi: 10.1186/s40649-

020-00080-x.
[17] B. Irena and Erwin Budi Setiawan, "Fake News (Hoax) Identification on Social Media Twitter using Decision Tree C4.5 Method," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 4, no. 4, pp. 711–716, 2020, doi: 10.29207/resti.v4i4.2125.
[18] N. Umar and M. Adnan Nur, "Application of Naïve Bayes Algorithm Variations On Indonesian General Analysis Dataset for Sentiment Analysis," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 4, pp. 585–590, 2022, doi: 10.29207/resti.v6i4.4179.
[19] W. Hidayat, E. Utami, and A. D. Hartanto, "Effect of Stemming Nazief Adriani on the Ratcliff/Obershelp algorithm in identifying level of similarity between slang and formal words," *2020 3rd Int. Conf. Inf. Commun. Technol. ICOIACT 2020*, pp. 22–27, 2020, doi: 10.1109/ICOIACT50329.2020.9331973.
[20] A. Aldoseri, K. N. Al-Khalifa, and A. M. Hamouda, "Re-Thinking Data Strategy and Integration for Artificial Intelligence: Concepts, Opportunities, and Challenges," *Appl. Sci.*, vol. 13, no. 12, 2023, doi: 10.3390/app13127082.
[21] J. G. K. Mabunda, A. Jadhav, and R. Ajoodha, "Sentiment Analysis of Student Textual Feedback to Improve Teaching," *Interdiscip. Res. Technol. Manag.*, pp. 643–651, 2021, doi: 10.1201/9781003202240-100.
[22] M. Avinash and E. Sivasankar, "A study of feature extraction techniques for sentiment analysis," *Adv. Intell. Syst. Comput.*, vol. 814, pp. 475–486, 2019, doi: 10.1007/978-981-13-1501-5_41.
[23] R. Obiedat *et al.*, "Sentiment Analysis of Customers' Reviews Using a Hybrid Evolutionary SVM-Based Approach in an Imbalanced Data Distribution," *IEEE Access*, vol. 10, pp. 22260–22273, 2022, doi: 10.1109/ACCESS.2022.3149482.
[24] C. A. Ul Hassan, M. S. Khan, and M. A. Shah, "Comparison of machine learning algorithms in data classification," *ICAC 2018 - 2018 24th IEEE Int. Conf. Autom. Comput. Improv. Product. through Autom. Comput.*, no. September, pp. 1–6, 2018, doi: 10.23919/IConAC.2018.8748995.
[25] W. van Atteveldt, M. A. C. G. van der Velden, and M. Boukes, "The Validity of Sentiment Analysis: Comparing Manual Annotation, Crowd-Coding, Dictionary Approaches, and Machine Learning Algorithms," *Commun. Methods Meas.*, vol. 15, no. 2, pp. 121–140, 2021, doi: 10.1080/19312458.2020.1869198.
[26] M. Umer, I. Ashraf, A. Mehmood, S. Kumari, S. Ullah, and G. Sang Choi, "Sentiment analysis of tweets using a unified convolutional neural network-long short-term memory network model," *Comput. Intell.*, vol. 37, no. 1, pp. 409–434, 2021, doi: 10.1111/coin.12415.
[27] M. Boukes, B. van de Velde, T. Araujo, and R. Vliegenthart, "What's the Tone? Easy Doesn't Do It: Analyzing Performance and Agreement Between Off-the-Shelf Sentiment Analysis Tools," *Commun. Methods Meas.*, vol. 14, no. 2, pp. 83–104, 2020, doi: 10.1080/19312458.2019.1671966.
[28] J. H. Joloudari, A. Marefat, M. A. Nematollahi, S. S. Oyelere, and S. Hussain, "Effective Class-Imbalance Learning Based on SMOTE and Convolutional Neural Networks," *Appl. Sci.*, vol. 13, no. 6, 2023, doi: 10.3390/app13064006.