DOI: 10.22146/jnteti.v14i3.20473

Sentiment Analysis of IKD Application Reviews on Play Store Using Random Forest

Kelvin H[1], Erlin[2], Yenny Desnelita[1], Dwi Oktarina[1]

[1] Department of Information Systems, Faculty of Computer Science, Institut Bisnis dan Teknologi Pelita Indonesia, Pekanbaru 28127, Indonesia
[2] Department of Informatics Engineering, Faculty of Computer Science, Institut Bisnis dan Teknologi Pelita Indonesia, Pekanbaru 28127, Indonesia

[Received: 26 March 2025, Revised: 13 June 2025, Accepted: 24 July 2025]
Corresponding Author: Kelvin H (email: kelvin.h@student.pelitaindonesia.ac.id)

ABSTRACT — The rapid growth of digital applications in population administration services has increased the importance of sentiment analysis to understand user perceptions more deeply. This study focuses on the Digital population identity (Identitas Kependudukan Digital, IKD), a digital identity application developed by the Indonesian government. It aims to classify user reviews of the IKD application into positive, neutral, and negative sentiments using the random forest algorithm. The dataset consisted of 28,134 user reviews from the Google Play Store, including usernames, review texts, timestamps, and star ratings. The research stages included data preprocessing, labeling, handling missing values, and text processing (cleansing, tokenizing, stopword removal, and stemming). The data were divided into 80% training and 20% testing sets. The best-performing model used the parameters: *max_depth=None*, *max_features=log2*, *min_samples_leaf=1*, *min_samples_split=2*, and *n_estimators=300*, achieving an average accuracy of 83.78%. To address class imbalance, the synthetic minority oversampling technique (SMOTE) was applied, resulting in improved performance with an accuracy of 86.29%. Evaluation metrics before SMOTE showed 83.85% accuracy, 80.40% precision, 83.85% recall, and 81.73% F1 score. After SMOTE, precision increased to 81.22%, while accuracy and recall slightly decreased to 80.86%, with an F1 score of 81.03%. Furthermore, sentiment trend analysis using N-gram techniques (unigram, bigram, trigram) was conducted to identify frequently mentioned topics and user concerns. These insights support the research objective of guiding application improvements aligned with user needs and enhancing the overall digital service experience.

KEYWORDS — Sentiment Analysis, Text Classification, IKD Application, Random Forest, SMOTE.

I. INTRODUCTION

Population identity plays a vital role in enabling Indonesian citizens to access essential public services, including healthcare, education, social assistance, and legal documentation. In line with the digital transformation agenda, the Indonesian government has introduced Digital population identity (Identitas Kependudukan Digital, IKD), which is a mobile-based application managed by the Directorate General of Population and Civil Registration (Ditjen Dukcapil). IKD is a digital version of the National Identity Card (Kartu Tanda Penduduk, KTP) that facilitates access to personal data through a mobile application. IKD enhances efficiency and accessibility to both public and private services without requiring physical documents, while also promoting digital and economic inclusion. Utilizing advanced encryption technology, IKD ensures data security and offers various benefits, including streamlined service access, administrative efficiency, and equitable access for all citizens [1]. The term "IKD" is used throughout this paper to refer to this digital identification platform. IKD aims to improve administrative efficiency by reducing bureaucracy, streamlining processes, and enhancing data security. However, its implementation still faces several challenges, including low digital literacy among users, uneven infrastructure availability, complex activation procedures, and less intuitive application interfaces [2], [3].

Given these challenges, understanding user perception and feedback toward the IKD application becomes essential. This is where machine learning, particularly sentiment analysis, plays a critical role. Sentiment analysis is a computational technique used to identify and classify opinions expressed in text as positive, negative, or neutral. This method is commonly applied to understand public perception of products, services, or specific issues, and it supports strategic decision-making by providing insights derived from the detected sentiments [4]. Sentiment analysis, powered by algorithms such as random forest, helps identify patterns in user reviews and classify them into positive, neutral, or negative sentiments. Random forest is an ensemble learning method known for its robustness, high predictive performance, and capacity to handle large datasets. When combined with text mining techniques, it becomes a powerful tool for extracting insights from textual data [5]–[7].

Previous studies have demonstrated the effectiveness of various machine learning models for sentiment analysis. For instance, research on hotel reviews in Purwokerto using random forest achieved an accuracy of 87.23%, without implementing stemming [8]. Another investigation examined IKD application feedback using the naïve Bayes algorithm, achieving an accuracy of 85.06% [9]. In addition, a comparative study involving sentiment analysis on metaverse-related data showed that random forest, when combined with the synthetic minority oversampling technique (SMOTE), performed better than support vector machine (SVM), achieving an accuracy rate of 91% [10].

Despite these promising results, many existing studies either focus on different domains or do not consider data imbalance problems. Therefore, this study is necessary to bridge these gaps by offering a novel combination of techniques tailored specifically to the Indonesian public service context. This research contributes new value by utilizing a large-scale real-world dataset (28,134 reviews) collected directly from the IKD Play Store page, applying a combination of random forest and SMOTE to address class imbalance,

conducting fine-grained sentiment pattern analysis using N-gram models (unigram, bigram, trigram), and focusing on Indonesian language user reviews, enhancing the local relevance of the model. This study not only evaluates the performance of random forest in sentiment classification but also explores the most discussed issues and appreciated features of IKD based on user feedback. The insights gained are expected to inform government policy, guide application development, and support more inclusive and responsive digital public services in Indonesia.

II. SENTIMENT ANALYSIS ON DIGITAL IDENTITY APPLICATIONS

The development of sentiment analysis in the context of digital public services has garnered increasing academic attention, particularly in evaluating user feedback on mobile applications. As digital identity systems, like IKD, become more prevalent in Indonesia, understanding user sentiment is crucial for ensuring successful adoption and continuous improvement of such platforms. Several previous studies have explored sentiment classification using various machine learning techniques, textual preprocessing strategies, and performance evaluation metrics. This section reviews relevant works that form the foundation for this research, highlighting the methods, findings, and limitations of past studies that analyzed sentiment in digital identity applications or comparable e-government services.

A 2024 study analyzed user sentiment toward the IKD application on the Google Play Store, aiming to assess public response and provide insights for service improvement [9]. Using web scraping via Google Colab, 1,000 reviews were collected and labeled as positive or negative based on user ratings, followed by manual verification. Preprocessing steps included cleansing, case transformation, tokenization, stopword removal, and token length filtering. Term frequency-inverse document frequency (TF-IDF) was used for feature weighting, and SMOTE was applied to address class imbalance. The naïve Bayes algorithm was employed with a 90/10 train-test split per iteration. Evaluation using a confusion matrix showed 85.06% accuracy, 80.31% precision, and 92.89% recall. Most reviews (619) were negative, frequently containing keywords like "difficult," "please," "document," "open," and "digital," highlighting usability issues. Strengths of the study include effective handling of class imbalance and thorough preprocessing. Limitations involve the absence of stemming, reliance on a single classifier, and a small dataset, affecting result generalizability. The authors suggest using more algorithms and larger datasets in future research [9].

A 2023 study examined public sentiment toward the IKD application, a government initiative to replace physical ID cards with digital versions [11]. Data were collected from Facebook comments between February 16–March 10, 2023, using the keyword "Identitas Kependudukan Digital." The sentiment analysis pipeline included data crawling, preprocessing (cleansing, case folding, tokenization, normalization, filtering, and stemming using Python), and feature weighting via TF-IDF. Sentiment classification employed a SVM with a multiclass approach—positive, negative, and neutral. Evaluation used a confusion matrix with accuracy, precision, recall, and F1 score metrics, and a word cloud for visualizing frequent terms. Among 902 comments, 78.27% were negative, 12.97% neutral, and 8.76% positive. The SVM model achieved 77% accuracy. Common negative terms such as "data" and "project" suggested concerns over data security and skepticism about the IKD program. Strengths include the use of SVM and real-time Facebook data, along with comprehensive preprocessing. Limitations involve platform-specific data, possible selection bias, and moderate model accuracy, indicating opportunities for further optimization and broader data sources. [11].

A 2024 study analyzed sentiment in user reviews of the IKD application on the Google Play Store using the k-nearest neighbor (KNN) algorithm [12]. Aimed at classifying reviews into positive, negative, or neutral categories, the study assessed public perception of Indonesia's digital identity initiative. Reviews were collected via web scraping, then preprocessed through case folding, tokenization, cleansing, stopword removal, stemming, and sentiment labeling based on user ratings. Text was vectorized using TF-IDF to emphasize significant terms. The KNN classifier was evaluated using various values of k and training/testing splits. The best results were achieved with k = 13 and an 80:20 split, yielding an accuracy of 83%, precision of 82%, recall of 83%, F1 score of 81%, and micro area under the curve (AUC) of 0.92. While the model effectively detected positive and negative sentiments, it struggled with neutral sentiment—achieving only 0.50 precision, 0.01 recall, and 0.01 F1 score—due to class imbalance. Overall, 66.75% of reviews were negative, indicating significant user dissatisfaction. Strengths include the use of TF-IDF, comprehensive evaluation across data splits, and practical insights for developers. Limitations include poor performance on neutral sentiment, high computational cost of KNN on large datasets, and the lack of semantic understanding in text analysis [12].

Previous studies addressed data imbalance using naïve Bayes with SMOTE, SVM for multi-class sentiment classification, and KNN with varying k values. However, these studies have limitations such as reliance on a single model, platform-specific data (e.g., Facebook), and poor performance in classifying neutral sentiments due to class imbalance.

This study addresses those gaps through a more comprehensive approach. It employs the robust random forest algorithm, evaluated in two scenarios—with and without SMOTE—to assess the impact of data balancing. By using Google Play Store reviews, the study captures real-world user sentiment more broadly. The preprocessing pipeline includes advanced normalization and N-gram analysis, enabling deeper insights into sentiment patterns. As a result, this research enhances both the methodological rigor and contextual relevance of sentiment analysis on the IKD application.

III. METHODOLOGY

This research follows a structured methodology consisting of several key stages to analyze user sentiment toward IKD application. The first stage was data collection, where user review data were obtained from the IKD application's page on Google Play Store. The data were collected using a crawling technique, an automated method for retrieving online information. The crawling process was implemented using the Python programming language combined with a tool called Google Playstore Scraper, which facilitated the extraction of relevant reviews posted by users under the developer account gov.dukcapil.mobile_id. The data collected included several key variables: username, review, date, and score. The username represents the reviewer's identity, review is the content of the user's feedback, date indicates when the review was posted,

and score reflects the user's rating of the app on a scale from 1 to 5. This method allowed for efficient and systematic data gathering to support subsequent analysis.

The next stage was data preprocessing, which involves several sub-processes to ensure data quality and readiness for machine learning modeling. Machine learning is a branch of artificial intelligence designed to address complex problems through three main approaches: supervised learning, which utilizes labeled data to enable accurate prediction or classification; unsupervised learning, which clusters unlabeled data to uncover hidden patterns or structures; and reinforcement learning, which makes optimal decisions based on feedback from the environment through exploration and evaluation. These approaches continue to evolve with the aim of enhancing computational efficiency and improving accuracy and precision in various applications [13]. The first step was data labeling, where domain experts manually evaluated each review to assign sentiment labels: positive, negative, or neutral. Neutral reviews, often consisting of spam or irrelevant content, were excluded from the dataset to maintain focus on meaningful sentiment expressions. Following labeling, text processing was conducted, which included cleaning the text, tokenization, removal of stopwords, stemming to reduce words to their base forms, and handling missing values. This step ensures that the textual data are standardized and free from noise. Once preprocessing was complete, data partitioning was performed by splitting the dataset into two subsets: 80% for training and 20% for testing. The training set was used to teach the model to recognize sentiment patterns, while the testing set was used to evaluate the model's performance on unseen data, ensuring its ability to generalize.

TF-IDF was applied to convert the preprocessed textual data into a format suitable for machine learning. It is a weighting method used in text processing to emphasize significant terms within a document while downscaling the weight of commonly occurring words. By transforming textual data into a numerical representation, TF-IDF enhances classification accuracy and is widely utilized in sentiment analysis to identify words that reflect positive or negative opinions [14]. This technique assigns weights to words based on their frequency within individual documents and their rarity across the entire dataset. TF-IDF helps emphasize terms that are important to a specific review while down-weighting commonly occurring terms that may carry less discriminative power. This numeric representation of text is essential for training classification algorithms effectively.

The core of the analysis lies in random forest modeling, which involves training an ensemble learning model known for its robustness and accuracy. Random forest is a machine learning technique that combines multiple decision trees through majority voting to perform classification. It is known for its high accuracy and robustness against overfitting. Compared to methods such as SVM and naïve Bayes, random forest offers more stable and reliable performance, making it well-suited for handling complex and diverse datasets [15]. The modeling process was refined using grid search to identify the optimal hyperparameters and was applied both to the original and the SMOTE-augmented dataset. SMOTE was used to balance class distribution, particularly in cases where there was a dominance of one sentiment class over others, which can negatively impact model performance. SMOTE is an oversampling method designed to address class imbalance issues in training data for classification algorithms [16].

Random forest testing evaluated the model's effectiveness. This involved comparing the performance of the model before and after applying SMOTE using the test dataset. The comparison aimed to assess whether balancing the dataset improved the model's predictive accuracy, precision, and recall across different sentiment categories.

Finally, the research included an N-gram analysis to gain deeper insights into the linguistic patterns of user sentiment. N-gram is a technique in NLP that segments text into sequences of adjacent items, such as unigram (n = 1), bigram (n = 2), and trigram (n = 3). This method enhances the understanding of context and sentence structure, thereby improving accuracy in text classification and various other NLP applications [17]. Unigram, bigram, and trigram models were employed to identify and quantify the most frequently occurring word sequences within each sentiment category. This analysis helps to reveal common expressions and concerns in user feedback, supporting a more nuanced interpretation of sentiment beyond numeric classification.

A. DATA COLLECTION

The data used in this study were obtained from the Google Play Store platform, serving as the primary source of user reviews related to IKD application. The Google Play Store not only provides information about the application but also contains reviews that reflect user experiences and perceptions, making it a relevant data source for sentiment analysis. IKD is available on the Play Store as an Android application, version 1.2.2, last updated on May 24, 2023. Initially released on June 3, 2022, by the Directorate General of Population and Civil Registration of the Ministry of Home Affairs, the application has a file size of 22 MB and is compatible with Android version 5.0 and above. It has been downloaded over 10 million times and is rated suitable for users aged 3 and above [18].

This study collected a total of 28,134 reviews through a crawling process. The extracted data included key elements such as username, review text, review date, and user-provided rating scores ranging from 1 to 5. The distribution of review scores highlights significant polarization in user opinions.

Reviews with a score of 1, indicating the highest level of dissatisfaction, accounted for 12,648 reviews or approximately 44.94% of the total data. These results suggest that many users are dissatisfied with the IKD application. Conversely, reviews with a score of 5, reflecting the highest level of satisfaction, amounted to 11,421 reviews or 40.60% of the total reviews. These figures indicate that, despite the significant number of negative reviews, the IKD application also received appreciation from a substantial number of satisfied users.

Reviews with scores of 2, 3, and 4 had smaller distributions. Specifically, score 2 accounted for 1,762 reviews (6.26%), score 3 for 1,352 reviews (4.81%), and score 4 for 951 reviews (3.38%). The dominance of extreme scores (1 and 5) suggests that user perceptions of the IKD application tend to be either very positive or very negative, with fewer reviews falling in the moderate range. This collected review data forms a crucial foundation for sentiment analysis, providing insights into user experiences and public perceptions of the IKD application.

B. DATA PREPROCESSING

Data Preprocessing is a crucial step in data processing that aims to enhance data quality before it is used for analysis or model training. This step transforms raw data into a more

structured and cleaner format, enabling efficient processing and improving model performance. Preprocessing plays a vital role in removing irrelevant information, addressing data inconsistencies or deficiencies, and preparing relevant features for subsequent analysis. In this study, the data preprocessing stage includes the following steps.

1) LABELING

In this step, each user review was assigned a sentiment label based on the given rating. High ratings (4–5) are classified as positive sentiment, while low ratings (1–2) are categorized as negative sentiment. Ratings with a score of 3 are classified as neutral. This process aims to provide a target variable for sentiment classification.

Out of the total analyzed data, there were 11,522 negative reviews; 1,113 neutral reviews; and 11,251 positive reviews. This distribution indicates that negative and positive reviews are nearly balanced in number, while neutral reviews are significantly fewer. This suggests that users tend to express strong opinions, either in the form of appreciation or complaints, rather than providing neutral or unbiased feedback.

2) HANDLING MISSING VALUES

Missing value handling involves detecting invalid or missing entries in the review text column. Functions, such as isnull() and the str.strip(), method are used to identify NaN values or empty strings. Invalid entries, such as empty text or strings containing only whitespace, are removed using the dropna() method and additional filters. After cleaning, the number of missing values was recalculated to ensure no empty values remain. This process guarantees that the dataset is fully valid, with the final results showing zero missing values.

3) TEXT PREPROCESSING

This process was carried out to refine and preprocess the text data before further analysis. It involves several key stages. First, data cleansing is performed to remove special characters, numbers, punctuation marks, and other irrelevant elements. Next, tokenization breaks down the review text into individual word units to facilitate structured analysis. Following this, stopword removal is applied to eliminate commonly used words that do not contribute significant meaning, such as “the,” “and,” or “is.” The final step was stemming, which utilized the Sastrawi library to reduce words to their root forms; for example, the word “running” would be transformed into “run.” These steps collectively produce clean and structured text data, ready for subsequent feature extraction and classification.

4) DATA PARTITIONING

The dataset was segmented into two portions: 80% for training and 20% for testing. Out of a total of 28,134 data points, approximately 22,507 were used for training, while 5,627 were allocated for testing. This partitioning ensures the model has sufficient data for training while also providing separate data to evaluate model performance.

From a total of 28,134 data points, 22,507 were utilized for model training (training set), while 5,627 were designated for evaluating the model’s performance (test set). This division ensures that the model has an ample amount of data to learn patterns from user reviews while maintaining a separate dataset to assess its accuracy in sentiment classification.

5) TF-IDF

This process transforms text data into numerical representations in the form of a sparse matrix, namely X_train_tfidf for training data and X_test_tfidf for testing data. The matrix includes 10,000 features that represent words in the dataset with weights based on their importance. The generated TF-IDF matrix is prepared to serve as input for machine learning models in the stages of sentiment analysis and classification.

C. RANDOM FOREST MODELING

The random forest modeling phase aims to construct an optimal classification model for analyzing user sentiment regarding IKD application. The modeling process involved two main approaches: without SMOTE implementation and with SMOTE implementation on the training data. Random forest is a classification method that constructs multiple decision trees using randomly sampled data. The process involves three main stages: bootstrap sampling (random sampling with replacement), random subsetting (random selection of features when building each tree), and aggregating (combining the predictions from all trees to determine the final output). This method is effective in handling non-linear data and identifying the most relevant variables in complex data analysis [19].

In the modeling approach without SMOTE, the model was constructed using grid search alongside a cross-validation strategy (k = 5) to determine the optimal hyperparameter configuration. The refined parameters included the number of trees (*n_estimators*), the maximum depth of decision trees (*max_depth*), the minimum sample count required for a split (*min_samples_split*), the minimum sample count for leaf nodes (*min_samples_leaf*), and the maximum number of features (*max_features*). The grid search method assessed a total of 288 parameter combinations, leading to 1,440 model evaluations through 5-fold cross-validation. The final outcome of this process was a set of optimized parameters that deliver the highest average accuracy on the training dataset.

In the modeling strategy, SMOTE was applied to address class imbalance by generating artificial samples for the underrepresented class, resulting in a more balanced training dataset. The resampled data were then used to build a random forest model with grid search and cross-validation. By improving class distribution, SMOTE enabled the model to better learn patterns in the minority class, enhancing its classification performance. This approach aims to produce more accurate and balanced results, with improved performance compared to models trained without SMOTE.

D. RANDOM FOREST TESTING

The random forest evaluation phase is designed to assess the model’s effectiveness in categorizing the test data. This assessment was performed after training the model under two conditions: with and without the application of SMOTE. The evaluation process utilized key performance metrics, including accuracy, precision, recall, and F1 score. Accuracy indicates the overall correctness of classifications, precision measures the reliability of positive predictions, recall determines the model’s sensitivity to positive reviews, and the F1 score offers a comprehensive balance between precision and recall. To measure the values of accuracy, precision, recall, and F1 score, it is essential to map the classification results into the categories of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) by comparing the predicted sentiment against the actual sentiment using a confusion matrix. A confusion matrix is an evaluation table that presents the number of correct and incorrect predictions made by a classification model relative to the actual values. It consists of four key

components: TP, TN, FP, and FN. This matrix is instrumental in calculating evaluation metrics such as accuracy, precision, and recall, which are crucial for assessing the performance and reliability of the classification model [20].

The evaluation phase was conducted to assess and compare the model's performance across both scenarios: without SMOTE implementation and with SMOTE implementation. This step was designed to determine the impact of the SMOTE technique in improving the random forest model's ability to handle class imbalances while ensuring its effectiveness in sentiment analysis on IKD application review data.

IV. RESULTS AND DISCUSSION

A. DATA COLLECTION

The data collection process was carried out using the scraping method with the Python programming language and the Google Play Scraper library. A total of 28,134 reviews were collected, covering a wide range of user opinions about their experience using the application. Each review contained information such as the username, review text, review date, and the score (star rating) given by the user, ranging from 1 to 5.

Based on the score distribution (Figure 1), the review data is dominated by two major groups: score 1 and score 5. Reviews with a score of 1, indicating the highest level of dissatisfaction, account for 12,648 entries or approximately 44.94% of the total dataset. This suggests that a significant portion of users had a poor experience with the IKD application. On the other hand, reviews with a score of 5, representing the highest level of satisfaction, are also substantial totaling 11,421 reviews or about 40.60% of the data. This indicates that despite many users being dissatisfied, there is also a large number of users who highly appreciated the app.

The remaining score groups were much smaller in comparison. Reviews with a score of 2 made up 1,762 entries (around 6.26%), score 3 accounted for 1,352 reviews (about 4.81%), and score 4 comprised 951 reviews (approximately 3.38%). This distribution reveals that user opinions are highly polarized, with most users either having a very positive or very negative experience, while moderate ratings (scores 2, 3, and 4) are relatively rare.

B. DATA PREPROCESSING

During the preprocessing stage, each review was assigned a sentiment label—positive, negative, or neutral—based on its rating. High ratings (4–5) were assumed to reflect positive sentiment, while low ratings (1–2) were considered negative. Ratings with a score of 3 were categorized as neutral. The goal of this labeling step was to prepare the data for use as a target variable in the sentiment classification process.

Based on the labeling results as shown in Figure 2, negative sentiment (labeled as -1) accounted for 14,410 reviews, indicating that approximately 51.2% of the analyzed data reflected user dissatisfaction with the IKD application. In contrast, positive sentiment (labeled as 1) was found in 12,372 reviews, or about 44% of the total dataset, showing that a significant number of users had a favorable experience. Neutral reviews (labeled as 0) totaled only 1,352 entries, or roughly 4.8%. These neutral reviews typically lacked strong emotional content and often contained irrelevant comments, links, or spam, making them less useful for further analysis.

This sentiment distribution reveals a tendency toward polarized opinions among users, with the majority expressing either strong satisfaction or dissatisfaction, and very few

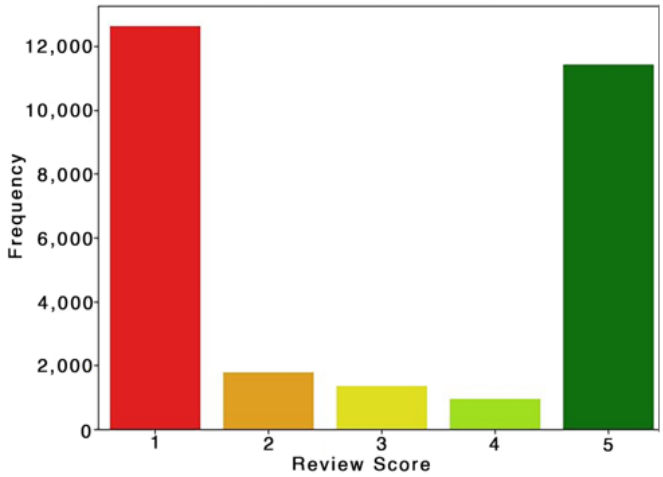

Figure 1. Dataset distribution by score.

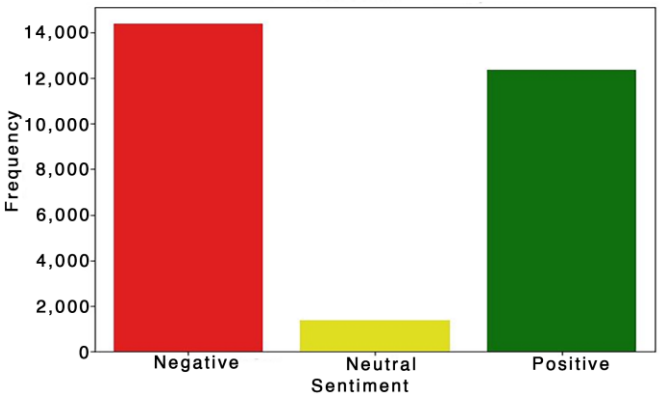

Figure 2. Sentiment labeling results.

remaining neutral. The labeling process enables a deeper analysis of the factors driving user sentiment, which can be instrumental in understanding public perception of the application and offering actionable insights for future development.

C. RANDOM FOREST MODELING

1) PARAMETERS USED

At this stage, the random forest model was built using a grid search technique with 5-fold cross-validation. Based on the experiments, the best-performing model was obtained with the following parameters: unlimited depth (*max_depth=None*), maximum features set to log2, a minimum of 1 sample per leaf, a minimum of 2 samples to split a node, and a total of 300 estimators (trees). This optimal model achieved an average accuracy of 83.78%. Given this result, the parameter combination {'*max_depth*': None, '*max_features*': '*log2*', '*min_samples_leaf*': 1, '*min_samples_split*': 2, '*n_estimators*': 300} was selected as the best configuration due to its highest average accuracy score.

2) SMOTE IMPLEMENTATION

In classification tasks, class imbalance in the training data is a common issue, where the number of samples in one class is significantly lower than in others. This imbalance can negatively impact model performance, especially in terms of accuracy and generalization ability. To address this problem, SMOTE is used to increase the number of samples in minority classes by generating synthetic data points.

Figure 3 shows the class distribution before SMOTE is applied. The dataset is clearly imbalanced, with 11,531 samples in the negative class, 9,774 in the positive class, and only 1,054 in the neutral class. The significant underrepresentation of the neutral class can lead to biased predictions, as the model may struggle to accurately identify or classify data from this minority group. Applying SMOTE helps balance the dataset, allowing the classification model to perform more fairly and reliably across all classes.

After initialization, SMOTE was applied to the training data (X_train_tfidf and y_train). This process produced two new variables: *X_train_resampled* (containing the resampled features) and *y_train_resampled* (containing the corresponding labels). SMOTE works by identifying underrepresented classes and generating new synthetic samples based on the existing data. Following the application of SMOTE, the class distribution in *y_train_resampled* was examined to confirm that each class had an equal number of samples. This step involved counting the number of instances for each class label within *y_train_resampled*.

As a result, the number of samples for each class (-1 for negative, 0 for neutral, and 1 for positive) in the training dataset became balanced, each with 11,531 samples (Figure 4). This balancing is crucial to ensure that the model does not become biased toward the majority class and is capable of recognizing patterns across all sentiment categories. Ultimately, this is expected to improve both the model's accuracy and its ability to generalize to new, unseen data.

D. RANDOM FOREST TESTING

In this research, the testing procedure involved comparing the model's performance in two different scenarios: one without the application of SMOTE and the other incorporating SMOTE. The evaluation process considered key performance metrics such as accuracy, precision, recall, and F1 score to ensure the model's consistency and robustness across different data conditions. The purpose of this comparison was to analyze the impact of SMOTE on the model's ability to effectively manage class imbalances.

Table I shows the model evaluation results before and after applying SMOTE. The model evaluation before applying SMOTE indicated an accuracy of 0.838462, meaning the model was able to correctly classify approximately 83.85% of the reviews. The precision score was 0.803984, indicating that around 80.40% of the reviews predicted as positive were indeed positive. The recall for the model before SMOTE was 0.838462, indicating that the model successfully identified approximately 83.85% of the positive reviews. Prior to applying SMOTE, the F1 score was 0.817261, offering a well-balanced representation of precision and recall, with an overall performance of approximately 81.73%.

After applying SMOTE, changes in these metrics were observed. The model's accuracy slightly decreased to 0.808587, indicating a minor reduction in its ability to correctly classify reviews. However, the precision improved slightly to 0.812209, suggesting that the model became more accurate in classifying positive reviews. The recall dropped to 0.808587, meaning the model identified slightly fewer positive reviews compared to before. Finally, the F1 score following the application of SMOTE was 0.810345, showing a slight decline

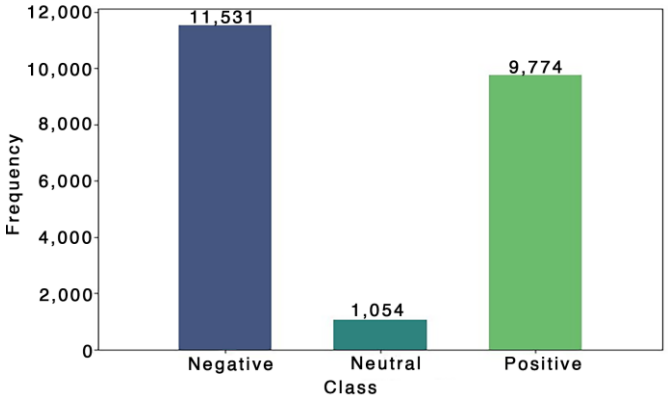

Figure 3. Class distribution before SMOTE.

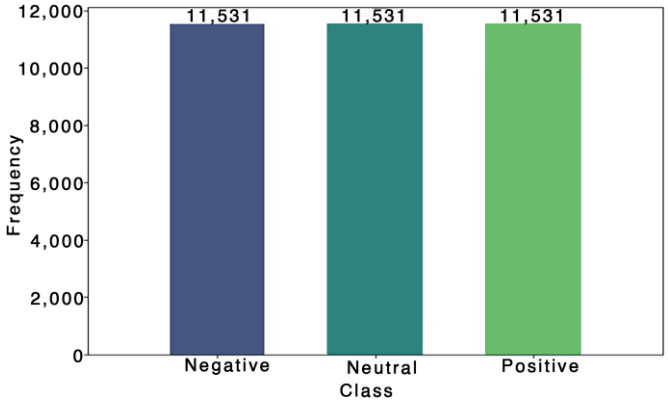

Figure 4. Sentiment distribution after SMOTE.

TABLE I
MODEL EVALUATION RESULTS

Metrics	Before SMOTE	After SMOTE
Accuracy	0.838462	0.808587
Precision	0.803984	0.812209
Recall	0.838462	0.808587
F1 score	0.817261	0.810345

compared to the previous value but still maintained a well-balanced trade-off between precision and recall.

E. ANALYSIS RESULT

This section presents the results of sentiment analysis conducted on user reviews of IKD applications on the Play Store. The analysis includes the performance of the random forest model, patterns of sentiment detected, and a deeper examination of user complaints and satisfaction levels.

The sentiment classification results were divided into three categories: positive, negative, and neutral reviews. For each category, emerging patterns in user feedback were observed to identify common themes.

Additionally, word analysis was carried out for each sentiment category using the N-gram method with values of n = 1, 2, and 3. N-gram is an NLP technique used to extract frequent word sequences in text.

1. Unigram (n = 1) analyzes individual words, allowing the identification of the most commonly used single terms in both positive and negative reviews.
2. Bigram (n = 2) examines frequent word pairs, helping uncover common phrases that describe specific user complaints or highlight the features most appreciated.

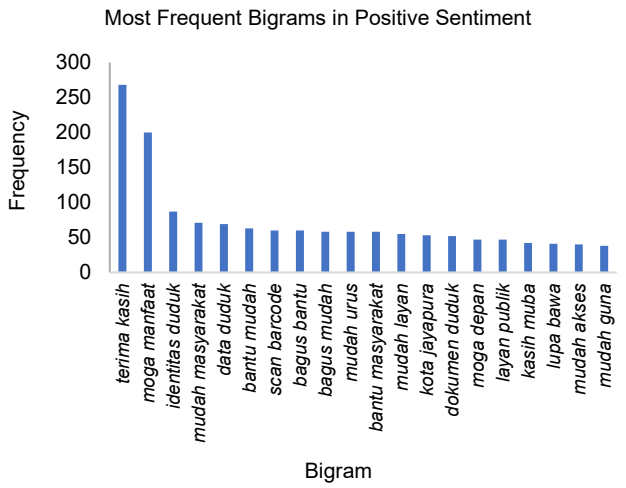

Figure 5. Word frequency based on bigrams for positive sentiment.

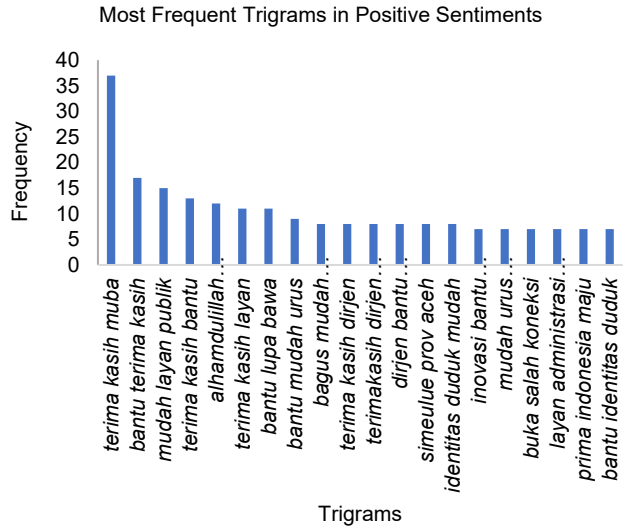

Figure 6. Word frequency based on trigrams for positive sentiment.

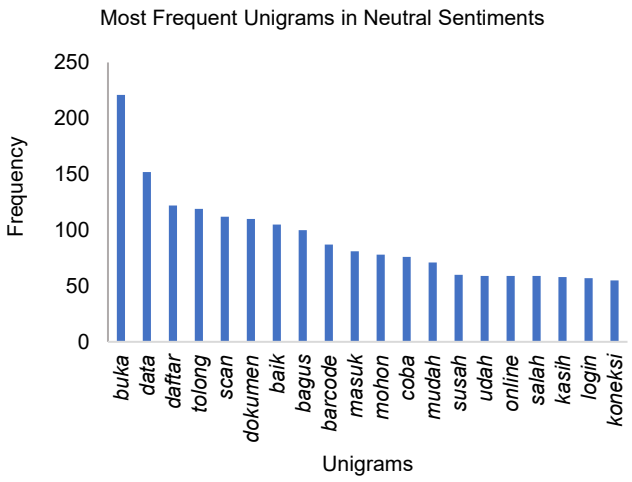

Figure 7. Word frequency based on unigrams for neutral sentiment.

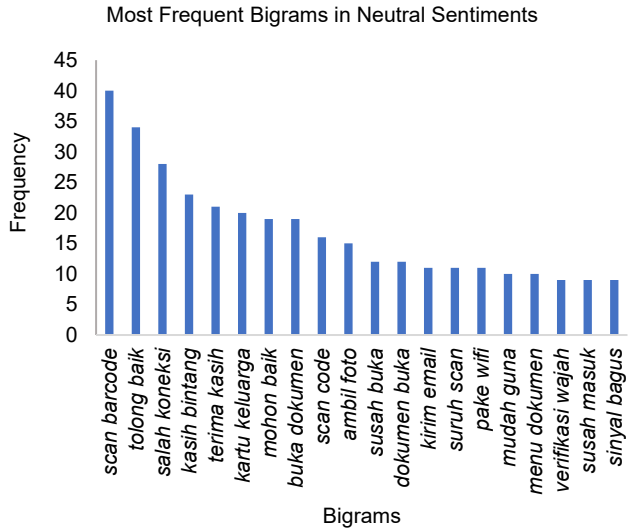

Figure 8. Word frequency based on bigrams for neutral sentiment.

3. Trigram (n = 3) focuses on sequences of three consecutive words, offering richer context into how users phrase criticism or praise.

By analyzing the most frequent words and phrases in user reviews, deeper insights can be gained into which aspects of the application need improvement and which features users find satisfying. This information is critical for developers aiming to make targeted enhancements based on direct user feedback. The results of this N-gram analysis will be presented in detail to support a clearer understanding of user perception toward the IKD application.

Figure 5 illustrates the results of the bigram implementation for positive sentiments. The bigram implementation for positive sentiment reveals combinations of words such as "*terima kasih*" (268 occurrences), "*moga manfaat*" (200 occurrences), and "*mudah masyarakat*" (71 occurrences), reflecting appreciation and the ease experienced by users. Phrases like "*identitas duduk*" (87 occurrences), "*data duduk*" (69 occurrences), and "*dokumen duduk*" (52 occurrences) highlight the ease of administrative services. Technological features, such as barcode scanning are appreciated, as seen in phrases like "*scan barcode*" (60 occurrences) and "*mudah akses*" (40 occurrences). Additionally, phrases like "*bagus bantu*," "*bagus mudah,*" and "*mudah guna*" indicate positive perceptions of service quality and ease of use.

Figure 6 illustrates the results of the trigram implementation for positive sentiments. The trigram implementation for positive sentiment shows phrases that reinforce user satisfaction, such as "*terima kasih muba*" (appearing 37 times), "*bantu terima kasih*" (17 occurrences), and "*mudah layan publik*" (15 occurrences), which reflect gratitude for the facilitative services. Phrases like "*alhamdulillah terimakasih dirjen*" (12 occurrences) and "*terima kasih dirjen*" (8 occurrences) indicate appreciation for the support of relevant authorities. Moreover, phrases like "*bantu lupa bawa*" (11 occurrences) and "*mudah urus administrasi*" (7 occurrences) highlight the practical aspects and ease of services, while "*inovasi bantu masyarakat*" (7 occurrences) illustrates the positive impact of innovations on society.

Figure 7 illustrates the results of the unigram implementation for neutral sentiments. The unigram implementation for neutral sentiment reveals words that reflect mixed user experiences. Words like "*buka*" (221 occurrences), "*data*" (152 occurrences), and "*daftar*" (122 occurrences)

indicate a focus on access and registration processes. Words such as "*tolong*" (119 occurrences) and "*mohon*" (appearing 78 times) suggest requests for assistance with technical obstacles. Terms like "*scan*" (appearing 112 times), "*barcode*" (appearing 87 times), and "*login*" (appearing 57 times) refer to the use of technology. Meanwhile words like "*susah*" (appearing 60 times), "*salah*" (appearing 59 times), and "*koneksi*" (appearing 55 times) highlight technical challenges such as errors or connectivity issues.

Figure 8 illustrates the results of the bigram implementation for neutral sentiments. The bigram implementation for neutral sentiment reveals mixed user experiences, blending satisfaction with technical difficulties. Phrases like "*scan barcode*" (40 occurrences) show frequent use, yet with potential challenges, while "*tolong baik*" (34 occurrences) and "*mohon baik*" (19 occurrences) reflect requests for smoother services. Bigrams like "*salah koneksi*" (28 occurrences) and "*susah buka*" (12 occurrences) indicate technical problems, particularly with access and login. Phrases such as "*kartu keluarga*" (20 occurrences) and "*verifikasi wajah*" (9 occurrences) highlight frequently used document verification processes, while "*pake wifi*" (11 occurrences) and "*sinyal bagus*" (9 occurrences) emphasize the importance of stable internet connections.

Figure 9 illustrates the results of the trigram implementation for neutral sentiments. The trigram implementation for neutral sentiment highlights technical issues experienced by users, especially related to connectivity and document access. Trigrams like "*scan barcode tugas*" and "*suruh scan barcode*" reflect confusion some users may experience with barcode scanning. Phrases like "*salah koneksi coba*" and "*buka salah koneksi*" point to connectivity issues hindering app usage. Trigrams like "*buka menu dokumen*" and "*dokumen kartu keluarga*" indicate interaction with important documents, while "*tanda tangan elektronik*" suggests verification procedures that require more attention. Overall, the results emphasize the need for improvements in the technical stability and accessibility of the app.

Figure 10 illustrates the results of the unigram implementation for negative sentiments. The unigram implementation for negative sentiment shows user complaints related to accessibility and technical issues. Words such as "*buka*" (2,216 occurrences), "*scan*" (1,222 occurrences), and "*barcode*" (1,050 occurrences) appear frequently, indicating problems with the scanning feature. Users also express dissatisfaction with registration and login processes with words like "*daftar*" (register), "*data,*" and "*masuk,*" while words like "*ribet*" (921 occurrences) and "*susah*" (848 occurrences) describe challenges. Technical issues, such as "*koneksi*" (appearing 804 times) and "*salah*" (appearing 746 times) and requests for fixes with words like "*tolong*" (740 occurrences) reflect user frustration with the app.

Figure 11 illustrates the results of the bigram implementation for negative sentiments. The bigram implementation for negative sentiment reveals primary user complaints about scanning and connectivity issues. Bigrams like "*scan barcode*" (554 occurrences) indicate difficulties with the scanning feature, supported by phrases such as "*suruh scan*" and "*scan code*." Connectivity issues are reflected in bigrams like "*salah koneksi*" (428 occurrences) and "*buka dokumen*" (134 occurrences), despite users having good networks, as indicated by phrases like "*jaring bagus*" and "*sinyal bagus*". Bigrams like "*mudah sulit*" and "*bikin ribet*" suggest features that frustrate users. Users also request fixes using phrases like "*tolong baik*" and "*mohon baik*," indicating hopes for improved stability and usability.

Figure 12 illustrates the results of the trigram implementation for negative sentiments. The trigram

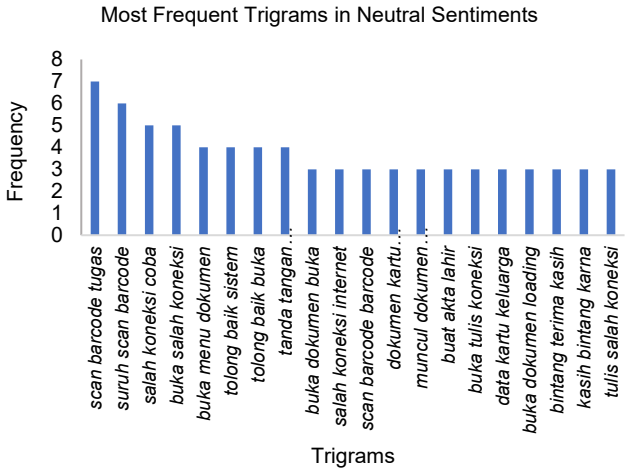

Figure 9. Word frequency based on trigrams for neutral sentiment.

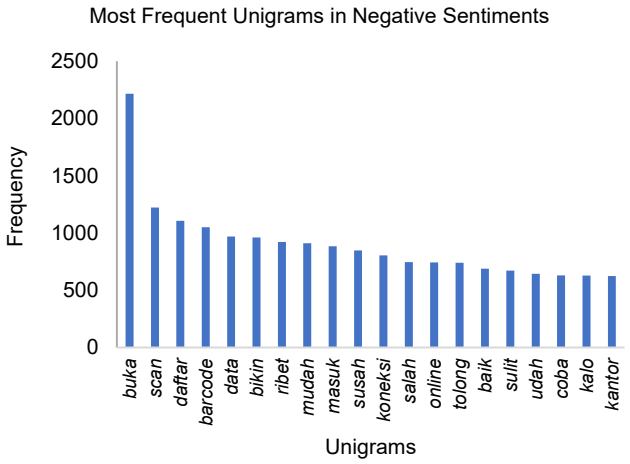

Figure 10. Word frequency based on unigrams for negative sentiment.

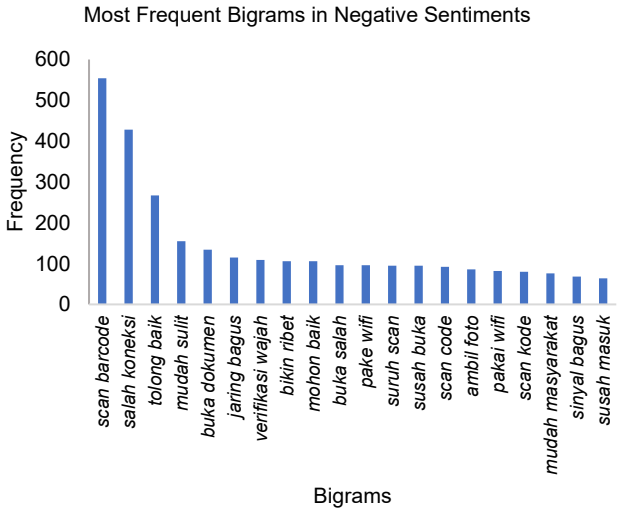

Figure 11. Word frequency based on bigrams for negative sentiment.

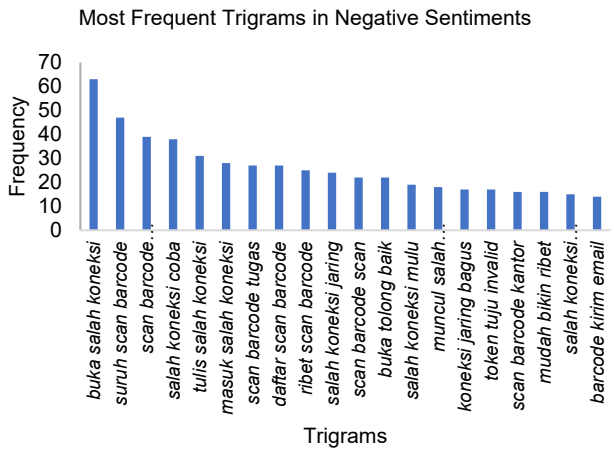

Figure 12. Word frequency based on trigrams for negative sentiment.

implementation for negative sentiment highlights key issues with connectivity and barcode scanning. Trigrams such as "*buka salah koneksi*" (63 occurrences) and "*salah koneksi coba*" (38 occurrences) reflect frequent complaints about network issues, while phrases like "*tulis salah koneksi*" (write wrong connection) and "*salah koneksi mulu*" (wrong connection all the time) indicate user frustration. In relation to barcode scanning, trigrams such as "*suruh scan barcode*" (ask to scan barcode) and "*ribet scan barcode*" (complicated scan barcode) point to difficulties with the scanning feature. Additionally, trigrams like "*token tuju invalid*" (invalid destination token) and "*barcode kirim email*" (barcode send email) suggest issues with authentication and data communication.

V. CONCLUSION

This study utilized 28,134 user reviews from the Google Play Store to classify sentiments toward the IKD application using the random forest algorithm. The model achieved 83.85% accuracy and an F1 score of 81.73% without class balancing. After applying SMOTE, precision improved, though accuracy and recall slightly declined, highlighting a trade-off between balance and generalization.

Compared to existing works using naïve Bayes, SVM, and KNN, the random forest model showed more consistent and competitive results, supported by extensive preprocessing and TF-IDF feature extraction. The model effectively handled imbalanced classes and diverse sentiment expressions in a large-scale dataset.

Additionally, N-gram analysis uncovered common themes in user feedback, such as ease of use and frustrations with login and connectivity. These insights demonstrate how machine learning can support user-centered improvements and contribute to the refinement of public digital services.

CONFLICTS OF INTEREST

The authors declare that there is no conflict of interest.

AUTHORS' CONTRIBUTIONS

Conceptualization, Kelvin H. and Erlin; methodology, Kelvin H and Erlin; software, Kelvin H; validation, Erlin; formal analysis, Kelvin H; investigation, Kelvin H; resources, Erlin, Yenny Desnelita, and Dwi Oktarina; data curation, Kelvin H; writing—original draft preparation, Kelvin H; writing—review and editing, Erlin, Yenny Desnelita, and Dwi Oktarina; visualization, Kelvin H; supervision, Erlin; project administration, Erlin, Yenny Desnelita, and Dwi Oktarina.

REFERENCES

[1] R.W. Sasongko, "Implementasi identitas kependudukan digital di Kabupaten Bandung," *J. Regist.*, vol. 5, no. 1, pp. 69–86, Sep. 2023, doi: 10.33701/jurnalregistratie.v5i1.3148.

[2] P.C. Ardilia, "Optimalisasi pendampingan layanan administrasi kependudukan melalui program Kalimasada untuk mewujudkan tertib adminduk studi di Kelurahan Tembok Dukuh Kota Surabaya," *PRAJA Obs., J. Penelit. Adm. Publik*, vol. 2, no. 3, pp. 63–68, May 2022.

[3] A. Widiyarta and I. Humaidah, "Implementasi aktivasi identitas kependudukan digital (IKD) dalam mendorong digitalisasi di Kelurahan Jepara Kota Surabaya," *J. Ilm. Wahana Pendidik.*, vol. 9, no. 18, pp. 43–51, Sep. 2023, doi: 10.5281/zenodo.8310255.

[4] M.N. Muttaqin and I. Kharisudin, "Analisis sentimen pada ulasan aplikasi Gojek menggunakan metode support vector machine dan k nearest neighbor," *UNNES J. Math.*, vol. 10, no. 2, pp. 22–27, Nov. 2021, doi: 10.15294/ujm.v10i2.48474.

[5] N.L.P.C. Savitri, R.A. Rahman, R. Venyutzky, and N.A. Rakhmawati, "Analisis klasifikasi sentimen terhadap sekolah daring pada Twitter menggunakan supervised machine learning," *JuTISI (J. Tek. Inform. Sist. Inf.)*, vol. 7, no. 1, pp. 47–58, Apr. 2021, doi: 10.28932/jutisi.v7i1.3216.

[6] A. Firdaus and W.I. Firdaus, "Text mining dan pola algoritma dalam penyelesaian masalah informasi: (Sebuah ulasan)," *JUPITER, J. Penelit. Ilmu Teknol. Komput.*, vol. 13, no. 1, pp. 66–78, Apr. 2021.

[7] A.C. Müller and S. Guido, *Introduction to Machine Learning with Python: A Guide for Data Scientists*. Sebastopol, CA, USA: O'Reilly Media, 2017.

[8] B.B. Baskoro, I. Susanto, and S. Khomsah, "Analisis sentimen pelanggan hotel di Purwokerto menggunakan metode random forest dan TF-IDF (Studi kasus: Ulasan pelanggan pada situs TRIPADVISOR)," *J. Inform. Inf. Syst. Softw. Eng. Appl. (INISTA)*, vol. 3, no. 2, pp. 21–29, May 2021, doi: 10.20895/inista.v3i2.218.

[9] A. Komarudin and A.M. Hilda, "Analisis sentimen ulasan aplikasi identitas kependudukan digital pada Play Store menggunakan metode naïve Bayes," *Comput. Sci. (CO-SCIENCE)*, vol. 4, no. 1, pp. 28–36, Jan. 2024, doi: 10.31294/coscience.v4i1.2955.

[10] P.K. Sari and R.R. Suryono, "Komparasi algoritma support vector machine dan random forest untuk analisis sentimen metaverse," *J. Mnemon.*, vol. 7, no. 1, pp. 31–39, Feb. 2024, doi: 10.36040/mnemonic.v7i1.8977.

[11] R.A. Lestari, A. Erfina, and W. Jatmiko, "Penerapan algoritma support vector machine pada analisis sentimen terhadap identitas kependudukan digital," *J. Teknol. Inf. Ilmu Komput.*, vol. 10, no. 5, pp. 1063–1070, Oct. 2023, doi: 10.25126/jtiik.20231057264.

[12] M. Ulfa, R.H. Kusumodestoni, and A. Sucipto, "Analisis sentimen review aplikasi identitas kependudukan digital di Google Play Store menggunakan KNN," *J. Inform. Teknol. Sains (JINTEKS),* vol. 6, no. 4, pp. 1155–1165, Nov. 2024, doi: 10.33558/jinteks.v6i4.4963.

[13] A. Roihan, P.A. Sunarya, and A.S. Rafika, "Pemanfaatan machine learning dalam berbagai bidang: Review paper," *IJCIT (Indones. J. Comput. Inf. Technol.)*, vol. 5, no. 1, pp. 75–82, May 2020, doi: 10.31294/ijcit.v5i1.7951.

[14] R. Wati, S. Ernawati, and H. Rachmi, "Pembobotan TF-IDF menggunakan naïve Bayes pada sentimen masyarakat mengenai isu kenaikan BIPIH," *J. Manaj. Inform. (JAMIKA)*, vol. 13, no. 1, pp. 84–93, Apr. 2023, doi: 10.34010/jamika.v13i1.9424.

[15] Erlin *et al.,* "Dampak SMOTE terhadap kinerja random forest classifier berdasarkan data tidak seimbang," *Matrik: J. Manaj. Tek. Inform. Rekayasa Komput.,* vol. 21, no. 3, pp. 677–690, Jul. 2022, doi: 10.30812/matrik.v21i3.1726.

[16] A. Mulianti, Y. Chrisnanto, and H. Ashaury, "Optimalisasi klasifikasi support vector machine dengan SMOTE: Studi kasus ulasan pengguna aplikasi Alfagift," *J. Pekommas*, vol. 9, no. 2, pp. 249–258, Dec. 2024, doi: 10.56873/jpkm.v9i2.5583.

[17] R. Chairunnisa, Indriati, and P.P. Adikara, "Analisis sentimen terhadap karyawan dirumahkan pada media sosial Twitter menggunakan fitur n-gram dan pembobotan augmented TF-IDF probability dengan k-nearest neighbour," *J. Pengemb. Teknol. Inf. Ilmu Komput.*, vol. 6, no. 4, pp. 1960–1965, Apr. 2022.

[18] *Identitas Kependudukan Digital.* (2025). Kementerian Dalam Negeri. Accessed: March. 10, 2025. [Online]. Available:

https://play.google.com/store/apps/details?id=gov.dukcapil.mobile_id&hl=id

[19] E. Christy and K. Suryowati, "Analisis klasifikasi status bekerja penduduk Daerah Istimewa Yogyakarta menggunakan metode random forest," *J. Stat. Ind. Komputasi*, vol. 6, no. 1, pp. 69–76, Jan. 2021.

[20] D. Normawati and S.A. Prayogi, "Implementasi naïve Bayes classifier dan confusion matrix pada analisis sentimen berbasis teks pada Twitter," *J-SAKTI (J. Sains Komput. Inform.)*, vol. 5, no. 2, pp. 697–711, Sep. 2021, doi: 10.30645/j-sakti.v5i2.369.