Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1017−1027
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.7763

Mental Health Sentiment Analysis on Twitter using Ensemble Learning Algorithm

Kemal Aziz*, Bambang Ari Wahyudi, Irma Palupi

School of Computing, Telkom University, Bandung, Indonesia
Email: [1,*]kstarid@student.telkomuniversity.ac.id, [2]bambangari@telkomuniversity.ac.id, [3]irmapalupi@telkomuniversity.ac.id
Correspondence Author Email: kstarid@student.telkomuniversity.ac.id
Submitted: **20/06/2025**; Accepted: **01/09/2025**; Published**: 02/09/2025**

Abstract−Mental health problems have become an important health issue around the world. Poor understanding as well as low mental health awareness contribute to mental health healing efforts. In particular, Social media is becoming a platform for people to convey feelings and emotions. A dataset of 20,000 English tweets, equally divided into 10,000 depressed and 10,000 non-depressed tweets, which were cleaned and processed using Term Frequency-Inverse Document Frequency (TF-IDF) for feature extraction. The method used in this sentiment analysis introduces an ensemble learning framework that combines Naïve Bayes, Support Vector Machine, and Random Forest classifiers, using majority voting for prediction. Each classifier was optimized using the best parameters, and the models were validated through 5-fold cross-validation. The experimental results show that Naïve Bayes with $\alpha = 1$ achieved an accuracy of 76.23% while Random Forest with 5000 trees at 76.77%, and Support Vector Machine with a linear kernel at 75.32%. By combining these classifiers, the ensemble model reached the highest accuracy of 77.88%, demonstrating the effectiveness of combining multiple models to improve performance.

Keywords: Ensemble Learning; Machine Learning; Mental Health; Sentiment Analysis; Social Media

1. INTRODUCTION

Mental health is a significant global issue that demands urgent attention, with the World Health Organization (WHO) estimating that one in four individuals will experience a mental health condition at some point in their lives [1]. Untreated mental health conditions, particularly depression, can lead to severe consequences such as suicide and self-harm, making early detection critical in mitigating these outcomes [2]. In this context, social media platforms such as Twitter (now known as X) have emerged as valuable sources of real-time, user-generated content that can offer valuable insights into public sentiment about mental health issues. These platforms generate vast textual datasets that, when analyzed, can provide transformative potential in identifying emotional patterns, detecting at-risk populations, and informing public health interventions [3]. Social media platforms like Twitter serve as modern-day journals, capturing the day-to-day sentiments and emotions of individuals from diverse demographic backgrounds. By analyzing the data, public health professionals and researchers can gain insights into societal attitudes toward mental health, thus enabling timely and targeted interventions.

Sentiment analysis, a key technique in natural language processing (NLP), is a systematic approach used to automatically classify subjective emotions, opinions, and attitudes expressed within textual data [4]. In the case of Twitter, sentiment analysis offers the possibility of detecting subtle emotional cues embedded in short messages, making it particularly useful for identifying individuals expressing mental health concerns. The rapid expansion of social media platforms has created vast amounts of unstructured data, which presents both opportunities and challenges. Traditional text analysis techniques often struggle to handle such large and varied datasets, requiring more sophisticated approaches such as machine learning to effectively process and interpret the information. Over the past decade, classical machine learning algorithms such as Support Vector Machines (SVM), Naïve Bayes (NB), and Random Forest (RF) have been applied extensively to tweet-level sentiment classification. These algorithms have shown considerable success in identifying the sentiment in social media posts, but they often encounter limitations when dealing with the complexities and nuances of human language.

Early research in the field of mental health detection focused primarily on extracting textual features from social media posts and applying machine learning algorithms such as Support Vector Machine (SVM) in year 2020 [5] and Random Forest in year 2019 [6] for sentiment classification. The research in year 2023 [7], applied the SVM algorithm to classify depression from approximately 10,000 comments on Facebook and YouTube, where SVM yielded the highest accuracy among tested algorithms, reinforcing its robustness in sentiment classification across different social media platforms. Similarly, the research in year 2022 [8] employed Random Forest alongside Word2Vec feature representations to identify depression symptoms on Twitter achieving 68.75% accuracy, indicating the approach's efficacy in supporting early mental health interventions via tweet analysis. While these early methods demonstrated the potential of machine learning in mental health detection, they were often limited by the inherent challenges of human language, especially in informal settings like social media. Sentences may be fragmented, contain abbreviations or slang, and may not follow the typical grammatical structures expected by many traditional machine learning models. These factors reduce the effectiveness of conventional algorithms in interpreting the sentiments expressed by users, especially those related to complex mental health conditions.

Sentiment analysis involves determining whether a text expresses a positive, negative, or neutral sentiment, and it is particularly useful in applications such as monitoring mental health on social media. In the context of mental health, sentiment analysis can help detect negative emotions such as depression or anxiety, allowing for the early

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1017−1027
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.7763

identification of individuals at risk. However, these tasks are not without challenges. Social media platforms, such as Twitter, contain a wealth of slang, emojis, hashtags, and misspellings, making it difficult for traditional models to identify emotional undercurrents. Moreover, a single post may convey a complex mixture of emotions, which can be hard to categorize using a simple positive-negative classification scheme. This complexity necessitates the use of more advanced techniques, particularly ensemble learning methods, to achieve more accurate and reliable results in sentiment classification.

While traditional machine learning methods like SVM and Random Forest have been widely applied to tweet classification, these models face challenges due to the informal, often slang-filled language found on social media platforms. Furthermore, these models may struggle to understand the complexities and subtleties inherent in human communication, which can make sentiment analysis on platforms like Twitter particularly difficult.

To address these challenges, ensemble learning has emerged as an effective solution. Ensemble learning is a method where multiple individual models (base models) are combined to create a stronger and more accurate predictive model [9]. By combining the strengths of several models, ensemble learning can overcome the limitations of single-model classifiers, particularly when applied to complex tasks like sentiment analysis. Popular ensemble techniques include bagging, boosting, and voting, which have shown to improve the performance of sentiment classification tasks [10]. Ensemble learning methods leverage the diversity of individual based models to achieve better generalization and higher accuracy.

The main advantage of ensemble methods is their ability to improve prediction accuracy by mitigating biases that may exist in individual models and reducing the risk of overfitting. By combining multiple models, ensemble techniques reduce the possibility of a single model making an error in the prediction, as the diversity of models leads to more reliable outcomes. In the context of mental health sentiment analysis on social media, ensemble learning is particularly advantageous, as it integrates the diverse strengths of classifiers like SVM, Random Forest, and Naïve Bayes. This integration helps to better capture subtle emotional cues and provides a more reliable solution for detecting mental health-related sentiments.

The primary objective of this research is to propose an ensemble learning framework that strategically combines SVM, Random Forest, and Naïve Bayes classifiers through an optimized ensemble methodology [11]. By employing systematic hyperparameter optimization and dynamic weight adjustment based on validation performance, the proposed framework aims to enhance the detection of subtle emotional expressions, improve overall classification accuracy, and offer a more robust solution for mental health sentiment analysis on Twitter data. This research seeks to tackle the complexities of detecting mental health-related sentiments in social media posts, improving upon previous single-model methods by creating a more accurate and generalizable ensemble model.

Furthermore, the research aims to contribute to the development of scalable, data-driven solutions for detecting mental health risks in real-time, which can be vital for timely interventions in public health. The outcomes of this study could potentially help public health agencies and mental health professionals identify at-risk individuals based on their online behavior, providing opportunities for early intervention and support.

2. RESEARCH METHODOLOGY

The system uses an ensemble learning approach to enhance the accuracy of sentiment analysis related to mental health issues. By integrating multiple machine learning models, it aims to provide more reliable and comprehensive results. The system design used in this research is illustrated in Figure 1. In general, the system involves several key stages, beginning with data preprocessing, feature extraction, training of base model classifiers, and concluding with model evaluation.

During the preprocessing stage, the available mental health dataset is prepared and cleaned using a series of techniques, including cleansing, case folding, tokenization, stopword removal, punctuation and number elimination, duplicate removal, emoticon replacement, and lemmatization. These processes are intended to clean and normalize the raw textual data so that it is ready for further processing and analysis. After the preprocessing stage is completed, the next step is feature extraction using the TF-IDF method to assign weights to words based on their frequency within the document and across the entire dataset.

After the feature extraction stage, the dataset is divided into training and testing sets. The training data is used to develop the best models. To ensure optimal performance and reliable evaluation, K-Fold Cross Validation is applied during the training process to create the best-performing models. The best models are built using several classification algorithms, including Naive Bayes, Support Vector Machine (SVM), and Random Forest. Upon obtaining the best model, the test data is used to evaluate its performance on new, unseen data. Predictions from the three classification models Naive Bayes, Support Vector Machine (SVM), and Random Forest are then combined using a voting method to decide the final class. This ensemble method helps the system produce more accurate and reliable results by using the majority decision from the individual models.

Finally, the system is evaluated using performance metrics derived from the confusion matrix, including accuracy, precision, recall, and F1-score. The performance results are then visualized to compare the effectiveness of the individual classifiers with that of the ensemble model, ensuring transparency and interpretability. This structured

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1017−1027
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.7763

workflow, integrating rigorous preprocessing, cross-validation, and ensemble learning, aims to deliver robust sentiment classification for mental health discourse on social media.

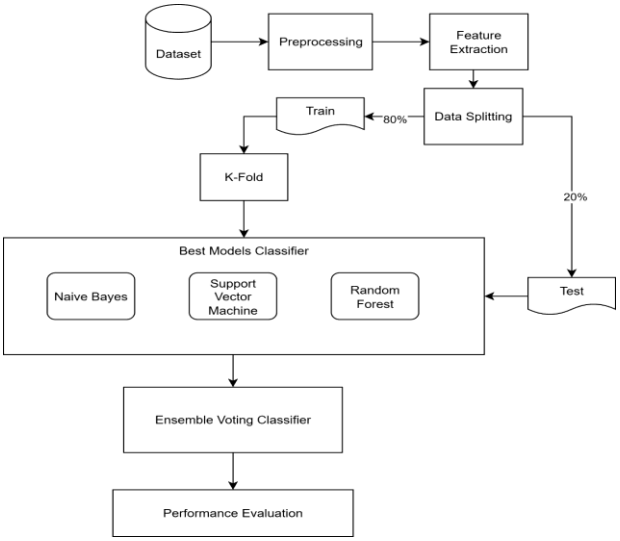

Figure 1. System design flowchart

2.1 Dataset and Preprocessing

The dataset used in this research consists of 20,000 English-language tweets, evenly split into 10,000 labeled as depressed and 10,000 labeled as non-depressed, representing two sentiment categories. These tweets were collected to analyze mental health-related sentiments expressed on social media [12].

Data preprocessing plays a critical role in the machine learning pipeline, as the quality of the data significantly influences the model's performance. The preprocessing steps in this research include cleansing, case folding, tokenization, stopword removal, and lemmatization. These techniques are applied to transform the raw data into a more structured and consistent format, enhancing the effectiveness of the subsequent analysis and model training.

2.1.1 Cleansing

Cleansing is the process of improving data quality by removing noise, inconsistencies, and irrelevant elements from raw text [13]. This includes eliminating URLs, user mentions, hashtags, emojis, and special characters. The process also entails correcting common misspellings and normalizing elongated words. These actions are critical for reducing noise and ensuring that the text is in a consistent format for subsequent analysis.

2.1.2 Case Folding

Case Folding is the process of converting text data sentences into uniforms. Case Folding is done by converting text into standard form, usually lowercase letters or also called lowercase. This step ensures that words like "Depression" and "depression" are treated identically, thereby reducing redundancy in the dataset and simplifying the feature space for machine learning models [14].

2.1.3 Tokenization

Tokenization is the process of segmenting text into individual units, typically words or terms, known as tokens. This is achieved by splitting the text based on whitespace and punctuation. Tokenization facilitates the analysis of text by enabling the examination of individual words and their frequencies within the corpus [15].

2.1.4 Stopwords Removal

Stopwords removal is the process of deleting words that are included in the stopword list [16]. Stopwords are common words that appear in large numbers that have a function but have no meaning to reduce noise and computational complexity. Removing stopwords helps focus the analysis on more meaningful words.

2.1.5 Lemmatization

Lemmatization data is a process to filter words that contain conjunctions, pronouns, prepositions, into basic words by eliminating prefixes or suffixes [17]. This process is essential for normalizing the text and ensuring that different forms of a word are analyzed as a single item, thereby improving the performance of sentiment analysis models.

2.2 Feature Extraction

Feature extraction is an important process that transforms unstructured textual data from mental health-related tweets into numerical representations suitable for machine learning models [18]. In this research, the Term Frequency-Inverse

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1017−1027
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.7763

Document Frequency (TF-IDF) is utilized as a feature extraction technique to determine the value of a specific word in a given document in contrast to other documents in the same set. This framework is commonly used to determine the weight of words in information retrieval and text mining applications.

Term Frequency (TF) measures how often a specific term appears in a document, normalized by the total number of words in that document. This normalization prevents bias toward longer documents, ensuring fair comparison across texts of varying lengths. Mathematically, TF is defined as formula 1.

$$TF = \frac{Number\ of\ occurrences\ of\ word\ in\ the\ document}{Total\ number\ of\ words\ in\ the\ document} \quad (1)$$

The second term component that will be useful in providing low weight to a phrase if it appears frequently in multiple documents is IDF, which stands for Inverse Document Frequency. The following IDF formula 2 determines how rarely a word appears in all documents.

$$IDF = log(\frac{Total\ number\ of\ documents}{Number\ of documents\ containing\ the\ word}) \quad (2)$$

The final TF-IDF score combines TF and IDF to reflect a term's importance within a document while accounting for its global rarity can be expressed using the formula 3.

$$TF - IDF = TF \times IDF \quad (3)$$

This weighting scheme ensures that terms with high frequency in a specific document but low frequency across the corpus receive the highest scores. For example, domain-specific terms such as "depression" or "self-harm" in mental health discussions often have elevated TF-IDF values, distinguishing them from common or generic vocabulary. By applying TF-IDF, the model can focus on meaningful words that contribute significantly to sentiment expression, while reducing the influence of frequently occurring but less informative words [19]. This enhances the effectiveness of the sentiment classification process on mental health-related tweets.

2.3 Data Splitting

To ensure a robust evaluation of the model's performance, the entire dataset, consisting of 20,000 samples, was partitioned into training and testing subsets using stratified random sampling. This method preserves the original class distribution while maintaining the proportion of depression and non-depression sentiment labels within both subsets, which is crucial for preventing bias during model training and evaluation.

Specifically, 80% of the dataset, amounting to 16,000 samples, was allocated for training, while the remaining 20%, consisting of 4,000 samples, was reserved for testing. The training subset was used to fit the model and learn the underlying patterns in the data, whereas the testing subset served to evaluate the model's ability to generalize and accurately classify unseen data [20].

By adopting this stratified splitting approach, the evaluation results reflect not only the model's capacity to capture meaningful patterns from the training data but also its effectiveness in making predictions on new, previously unseen instances. This balance is essential to assess the general applicability of the sentiment analysis model.

2.4 K-Fold Cross Validation

K-Fold cross-validation is a resampling procedure in which a dataset is partitioned into k equally sized folds. The model is trained and evaluated k time with each time using a different fold as the validation set and the remaining k–1 fold for training. By averaging performance metrics across all folds, one obtains a more reliable estimate of generalization error than from a single train/test split.

In this research, 5-fold cross-validation was employed during the training phase to ensure robust evaluation and mitigate overfitting. The training dataset of 16,000 samples was partitioned into five equal subsets (folds) of 3,200 samples each. In each iteration, four folds (12,800 samples, representing 80% of the training data) were used for training, and one fold (3,200 samples, representing 20% of the training data) was used for validation. This process was repeated five times, with each fold serving as the validation set exactly once, ensuring comprehensive utilization of the training data.

To maintain the original class distribution within each fold, stratified K-fold cross-validation was applied. This approach is particularly important for datasets with balanced classes, as it preserves the proportion of each class in every fold, thereby enhancing the reliability and representativeness of the evaluation metrics [21]. Additionally, the data was shuffled before splitting to reduce bias caused by any inherent ordering in the dataset.

2.5 Best Models Classifier

Machine learning approaches, particularly supervised learning algorithms, have been widely applied in sentiment classification tasks to categorize textual data such as tweets into predefined sentiment classes with varying degrees of accuracy. In this research, several base classifiers were selected due to their proven effectiveness and complementary strengths in handling text classification problems. The classifiers include Support Vector Machine (SVM), Random Forest (RF), and Naïve Bayes (NB), each contributing unique capabilities to address the linguistic complexities of mental health discourse on social media.

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1017−1027
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.7763

2.5.1 Naive Bayes (NB)

Naive Bayes is a simple yet powerful probabilistic classifier based on Bayes' theorem, which assumes that features are conditionally independent given the class label [22]. This assumption, while often violated in practice, allows for efficient computation and has made Naive Bayes a popular choice in text classification tasks such as sentiment analysis. It calculates the probability of each class given the input features and assigns the class with the highest probability.

In this research, the Multinomial Naive Bayes (MNB) variant was employed, which is particularly well-suited for discrete feature counts commonly found in text data. A key hyperparameter in MNB is the additive smoothing parameter, denoted as alpha. This parameter helps to handle the problem of zero probabilities for features that do not appear in the training samples of a given class.

Adjusting the alpha value helps balance the complexity and error. Higher alpha values increase the smoothing effect, which helps reduce overfitting by preventing the model from assigning zero probability to unseen features. However, if the alpha value is set too high, it can introduce bias by making the model overly simplistic. On the other hand, lower alpha values reduce smoothing, allowing the model to fit the training data more closely, but this can increase the risk of overfitting by capturing noise and details that don't generalize well to unseen data [23].

2.5.2 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a powerful supervised learning algorithm that aims to find the optimal hyperplane separating different classes by maximizing the margin between them. This margin maximization principle enables SVM to achieve strong generalization performance, especially in high-dimensional and sparse datasets such as text data. SVMs can handle both linear and non-linear classification problems by employing kernel functions that implicitly map input data into higher-dimensional feature spaces. Common kernel types include linear, polynomial, and radial basis function (RBF) kernels, with the choice of kernel significantly affecting the model's ability to capture complex patterns in the data [24].

Key hyperparameters in SVM include the regularization parameter C, kernel type, gamma (γ), and the option to enable probability estimates. The regularization parameter C controls the trade-off between maximizing the margin and minimizing classification errors: a smaller C encourages a wider margin but allows more misclassifications, increasing bias, while a larger C attempts to classify all training examples correctly but may lead to overfitting and higher variance. The kernel type determines the transformation applied to the input data, with the linear kernel suitable for linearly separable data and the RBF kernel capable of modeling complex, non-linear relationships [25].

Gamma, relevant for non-linear kernels like RBF, defines the influence of a single training example on the decision boundary, a low gamma value results in smoother boundaries by considering points farther from the boundary, whereas a high gamma value creates tighter, more complex boundaries that may overfit the training data. Enabling probability estimates allows the model to output class probabilities instead of just class labels, which requires additional internal cross-validation during training, increasing computational cost but providing richer prediction information.

2.5.3 Random Forest (RF)

Random Forest (RF) is a method that constructs multiple decision trees during training and aggregates their predictions to produce a final classification output. By combining the results of diverse trees built on random subsets of the training data and features, Random Forest effectively reduces the risk of overfitting that is common in individual decision trees, while enhancing overall accuracy and robustness [26].

The key hyperparameters considered in this research include the number of trees, maximum features, maximum depth, and class weight. The number of trees generally enhances model stability and accuracy by reducing variance, although it also increases computational cost and training time. The number of features to consider when looking for the best split. The maximum features, introducing randomness and diversity among the trees by limiting the number of features considered at each split, helps reduce correlation between individual trees and improves the model's generalization ability.

Maximum depth limits the maximum depth of each tree. If set to None, trees grow until all leaves are pure or contain fewer samples than the minimum required to split. Controlling tree depth helps prevent overfitting by limiting model complexity. Lastly the class weight associated to handle imbalanced datasets. Setting this to 'balanced' adjusts weights inversely proportional to class frequencies. The hyperparameter tuning process involved systematically exploring these parameter values to identify the configuration that maximizes classification performance on mental health-related tweets [27]. The best-performing Random Forest model obtained from this process was saved for subsequent evaluation and deployment.

2.6 Ensemble Voting Classifiers

Ensemble voting classifiers are powerful machine learning techniques that combine predictions from multiple individual models to enhance overall classification performance. The core idea behind ensemble methods is that aggregating diverse base classifiers can yield higher accuracy and robustness than any single model alone. This is

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1017−1027
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.7763

because different classifiers often capture distinct patterns or aspects of the data, and their combined decisions help mitigate errors caused by individual model biases or variances [28].

In this research, weighted majority voting was used to predict class probabilities from several base classifiers and assigned each input sample to the class with the highest average predicted probability. Unlike simple majority voting, which counts only the predicted class labels, weighted majority voting leverages the confidence of each classifier's predictions, often resulting in improved performance when base models provide well-calibrated probability estimates.

This ensemble approach is particularly beneficial for sentiment classification of mental health-related tweets, where linguistic nuances, varied expressions, and subtle contextual cues make the task challenging. The voting mechanism stabilizes predictions across diverse data samples, improving the system's ability to generalize and deliver more reliable sentiment analysis results.

2.7 Performance Evaluation

Performance evaluation is essential for assessing the effectiveness of machine learning models in accurately classifying sentiment. This evaluation utilizes key metrics such as accuracy, precision, recall, and F1-score, providing a comprehensive assessment of the model's performance. Additionally, the confusion matrix serves as a tool to summarize and analyze the model's predictions.

Table 1. Confusion Matrix

Confussion Matrix	Predicted Positive	Predicted Negative
Actual Positive	True Positive	False Positive
Actual Negative	False Negative	True Negative

Table 1 is an overview of the confusion matrix. The confusion matrix encompasses four key terms that aid in understanding model predictions. True Positive (TP), which refers to correctly predicted positive instances. True Negative (TN), indicating correctly predicted negative instances. False Positive (FP), representing incorrect predictions where the true label is negative but predicted as positive. False Negative (FN), which describes incorrect predictions where the true label is positive but predicted as negative. The following formulas were used to calculate the performance metrics.

a. Accuracy: The ratio of correctly predicted instances (both positive and negative) to the total number of instances. The following formula can be used to calculate accuracy:

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{4}$$

b. Precision: The ratio of correctly predicted positive instances to all instances predicted as positive. The following formula can be used to calculate precision:

$$Precision = \frac{TP}{(TP + FP)} \tag{5}$$

c. Recall: The ratio of correctly predicted positive instances to all actual positive instances. The following formula can be used to calculate recall:

$$Recall = \frac{TP}{(TP + FN)} \tag{6}$$

d. F1-Score: A harmonic mean of precision and recall, providing a balanced measure of both metrics. The following formula can be used to calculate F1-Score:

$$F1 - Score = \frac{(2 \times (Precision \times Recall))}{(Precision + Recall)} \tag{7}$$

These metrics together provide a comprehensive evaluation of the classification model. While accuracy offers a general sense of performance, it can be misleading in imbalanced datasets where one class dominates [29]. Precision and recall give more detailed insights into the model's performance on positive class predictions, which are particularly important in sentiment analysis tasks. The F1-score balances these two metrics, offering a unified measure that reflects both precision and recall. By analyzing the confusion matrix alongside these performance indicators, the robustness and reliability of the classification model can be effectively assessed, ensuring that it performs well overall.

3. RESULT AND DISCUSSION

In this research, the performance of sentiment analysis models for mental health-related tweets was systematically evaluated using an ensemble learning approach. This approach combined multiple base classifiers namely Naive Bayes, Support Vector Machine (SVM), and Random Forest (RF) to leverage their complementary strengths in classifying sentiments accurately. The primary goal was to enhance the overall prediction accuracy and robustness, particularly for detecting subtle emotional cues related to mental health in social media content.

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1017−1027
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.7763

3.1 Base Classifiers Performance

To evaluate the performance of individual classifiers, three widely used algorithms were implemented: Multinomial Naïve Bayes (MNB), Support Vector Machine (SVM), and Random Forest (RF). These models were trained and tested using consistent preprocessing steps and TF-IDF feature extraction. The evaluation employed 5-fold cross-validation and hyperparameter tuning to identify optimal configurations for each model.

3.1.1 Naive Bayes (NB)

The Multinomial Naïve Bayes (MNB) classifier was chosen for this research due to its strong track record in text classification tasks that involve discrete features. Its computational efficiency in handling high-dimensional natural language datasets is particularly beneficial when working with social media text data, where vocabulary size and data sparsity are key challenges. The MNB classifier operates under the assumption that the features (words) are conditionally independent, making it effective in scenarios where text data might be noisy, as is often the case with social media content.

To optimize the performance of the Naive Bayes classifier, hyperparameter tuning was performed using 5-fold cross-validation combined with GridSearchCV. The primary hyperparameter tuned was the smoothing parameter (α), which is crucial for preventing zero probabilities in cases where a word does not appear in the training data. The values tested for α were {1.0, 2.0, 3.0, 4.0}, as these represent a reasonable range of values for smoothing in natural language processing tasks.

Table 2. MNB Cross-Validation Performance.

Fold (%)	Alpha = 1.0	Alpha = 2.0	Alpha = 3.0	Alpha = 4.0
Fold 1	75.99	71.23	73.20	73.91
Fold 2	74.09	74.42	75.20	73.16
Fold 3	77.14	75.76	71.53	74.09
Fold 4	75.23	76.56	75.20	76.60
Fold 5	78.71	77.87	77.48	71.04
Accuracy (%)	76.23	75.17	74.53	73.76

The results from Table 2 show the performance of MNB across folds for various α values. In particular, the best performing classifier with α = 1.0 and 76.23% accuracy consistently achieved the highest accuracy across all five folds. This confirms the robustness of the model when using a smoothing factor of 1.0, which offers the best trade-off between bias and variance.

3.1.2 Support Vector Machine (SVM)

The Support Vector Machine (SVM) classifier was employed with a linear kernel for this research. SVM is a powerful classification algorithm that works by finding the hyperplane that best separates the data into distinct classes. The choice of using a linear kernel was driven by the high-dimensional, sparse nature of the TF-IDF feature space, which is common in text-based datasets. Linear kernels have proven effective in such settings, as they perform well when the data is linearly separable or can be separated with a linear decision boundary

To optimize the performance of the SVM classifier, hyperparameter tuning was conducted using 5-fold cross-validation combined with GridSearchCV. The hyperparameter search focused on identifying the best values for the penalty parameter (C), kernel type, and kernel coefficient (γ). The search space for these parameters included the following ranges: C ({0.1, 1, 10, 100}), kernel types ({linear, rbf}), and kernel coefficient γ ({scale, auto}). The optimization process allowed for identifying the optimal configuration that would provide the best trade-off between bias and variance, ensuring that the classifier generalized well to unseen data.

The results of this experiment are illustrated in Figure 2, which presents the average accuracy across different combinations of kernel types and C values. It is evident that the linear kernel consistently outperformed the RBF kernel across all tested values of C. The highest performance was achieved at C = 1, yielding an accuracy of 75.32%. Based on these findings, the configuration kernel = linear, C = 1, γ = scale was selected as the optimal model. This combination demonstrated the best trade-off between accuracy and stability while maintaining a simple and interpretable decision boundary.

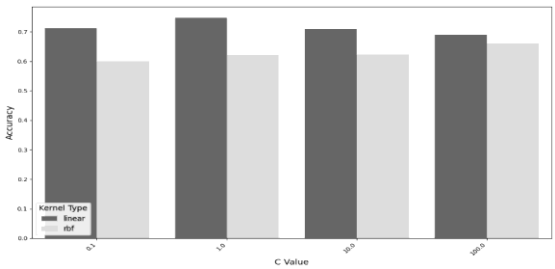

Figure 2. Comparison of average accuracy for SVM models

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1017−1027
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.7763

3.1.3 Random Forest (RF)

The Random Forest (RF) classifier was selected for its robust ensemble voting mechanism, which effectively reduces overfitting and handles class imbalance—common challenges in sentiment analysis of social media data. The RF model was configured with 5,000 decision trees, a maximum feature selection strategy of log2, and a class weighting scheme set to "balanced" to mitigate the impact of uneven class distributions in the dataset.

Hyperparameter tuning was conducted using 5-fold cross-validation, confirming that this configuration yielded the best performance with an accuracy of 76.77%. The use of a large number of trees enhances the model's stability and reduces variance, while the log2 feature selection strategy ensures that each tree considers a manageable subset of features, promoting diversity among trees and improving generalization.

3.3 Ensemble Learning Classifier Performance

The ensemble model, which utilized weighted majority voting, demonstrated enhanced performance in this research, especially when compared to individual classifiers. Weighted majority voting was particularly effective in situations where base classifiers output discrete class labels rather than probabilities, and it outperformed simple majority voting methods. Each base classifier cast a "vote" for a specific class, and the class that received the highest number of votes became the final prediction.

The ensemble model achieved an accuracy of 77.88%, which represents a notable improvement over single classifiers. This increase in performance underscores the ensemble model's ability to leverage the unique strengths of each base classifier, resulting in a more accurate detection of subtle emotional cues within the data. By combining the predictions of multiple classifiers, the ensemble method outperformed individual models, demonstrating its robustness and superiority in sentiment classification tasks.

To further illustrate, Figure 3 presents a comparison of sentiment distribution predicted by each model. Naïve Bayes displayed a tendency to predict more depressed sentiments, potentially contributing to a higher false-positive rate, while Random Forest leaned toward identifying more non-depressed sentiments, likely making it more conservative. The ensemble model balanced both sentiment classes more evenly, indicating a more calibrated and nuanced decision boundary, as it successfully harmonized the prediction biases of its base learners.

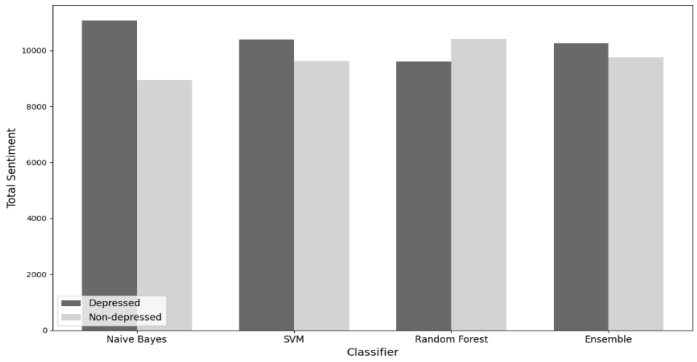

Figure 3. Comparison of predicted sentiment totals

3.3 Discussion

This research evaluated four classifiers namely Naïve Bayes, Support Vector Machine (SVM), Random Forest, and an Ensemble model on a mental health sentiment dataset. The key performance metrics considered were accuracy, precision, recall, and F1-score for both sentim ent classes. These metrics offer a comprehensive view of each model's ability to classify sentiments correctly.

Previous studies on mental health sentiment classification have frequently utilized traditional machine learning algorithms, including Support Vector Machine (SVM) with an accuracy of 75.15% and Random Forest with an accuracy of 68.75% [8]. However, this research aims to enhance classification performance through the use of ensemble learning framework that combines multiple classifiers.

Table 3. Classifier Performance Metrics for Sentiment Analysis.

Classifier	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Naive Bayes	76.23	80.03	82.35	78.48
Support Vector Machine	75.32	76.43	77.52	76.50
Random Forest	76.77	76.27	76.46	76.63
Ensemble	77.88	79.87	81.67	78.84

The experimental results comparing the ensemble learning model to individual classifiers are presented in Table 3. The ensemble model achieved the highest overall accuracy of 77.88%, outperforming Naïve Bayes (76.23%), Random Forest (76.77%), and Support Vector Machine (75.32%). Although the improvement is between 1-2%, this

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1017−1027
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.7763

increment highlights the advantage of combining multiple classifiers. The ensemble approach effectively leverages the strengths of individual models to deliver a more robust and reliable performance compared to any single classifier.

In order to further explore the results, Figure 4 presents the confusion matrices for each model. These matrices give a detailed overview of the true positive, true negative, false positive, and false negative rates. As seen in Naive Bayes, the model has relatively high false positive rates (14.5%), suggesting it may incorrectly classify non-depressed instances as depressed. On the other hand, the SVM and Random Forest classifiers demonstrate better balance in predicting both sentiment classes. However, the ensemble model consistently outperforms the individual models in terms of accuracy, precision, recall, and F1-score. This is particularly notable when comparing true positives and false negatives, where the ensemble model shows the best results.

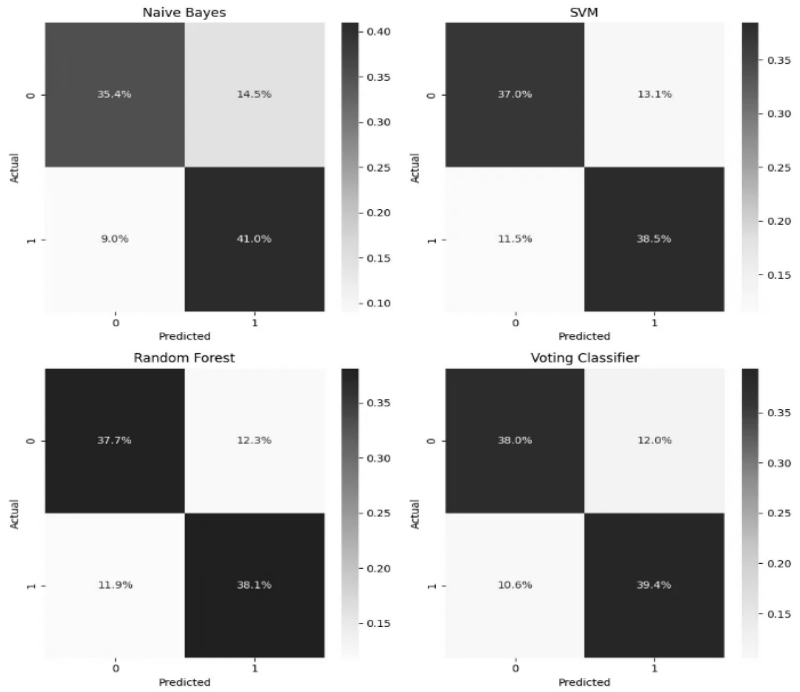

Figure 4. Confusion Matrices of Classifiers

Figure 5 displays a bar chart comparison of the precision, recall, and F1-score for all four classifiers. As shown in the chart, the Ensemble model consistently performs better across all metrics. It highlights the importance of using ensemble learning in sentiment analysis tasks, especially in the context of mental health detection on social media, where both precision and recall need to be optimized.

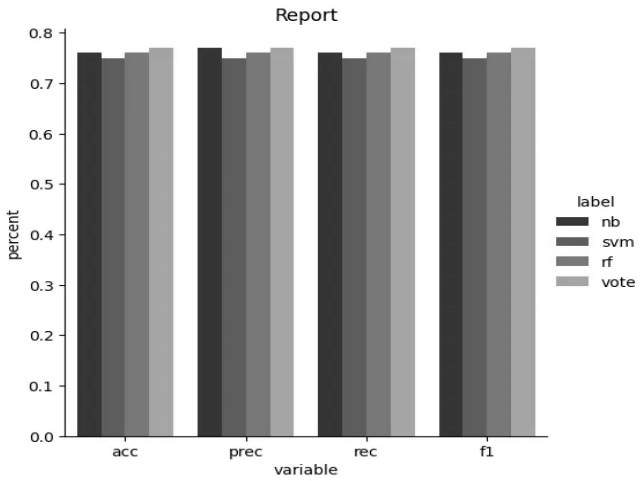

Figure 5. Performance Visualization of Classifiers

While the accuracy improvement from ensemble learning is relatively modest, this reinforces the concept that ensemble methods are particularly effective for complex tasks like sentiment analysis, where the detection of nuanced emotions in social media posts is critical. The ensemble approach not only provides a better balance between false positives and false negatives but also ensures that all classifiers contribute their unique strengths to the final prediction.

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1017−1027
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.7763

These results confirm that combining multiple models can significantly enhance mental health sentiment analysis on social media, particularly when detecting subtle emotional cues that may otherwise be overlooked by single classifiers.

However, it is important to note that the improvement gained from ensemble learning is directly influenced by the performance gap between the individual classifiers. In cases where the classifiers perform similarly, the ensemble model will reflect these small variations, leading to modest improvements. For example, in this study, the accuracy difference between models was small, which resulted in only a slight increase in performance when combined. These findings reinforce the idea that ensemble learning, especially with diverse classifiers, can outperform individual models in predicting sentiments, making it a highly effective tool for mental health sentiment analysis. Despite the modest improvement in accuracy, the ensemble model demonstrates its potential for broader applications in detecting mental health conditions through social media.

4. CONCLUSION

This research explored the effectiveness of combining three machine learning classifiers Naïve Bayes, Support Vector Machine (SVM), and Random Forest—into an ensemble model for mental health sentiment analysis on Twitter data. The ensemble approach significantly improved classification performance, achieving the highest accuracy of 77.88%, with balanced precision, recall, and F1-scores across both sentiment classes. This performance improvement resulted from the ensemble's ability to leverage the complementary strengths of individual classifiers, effectively capturing different aspects of sentiment analysis. As a result, the ensemble model was able to provide more accurate and reliable detection of mental health-related sentiments, which is crucial for real-time applications like monitoring mental health trends on social media platforms. The findings demonstrate that ensemble learning methods can significantly enhance sentiment classification, especially in the complex domain of mental health detection on social media. This research highlights the potential of ensemble models to address difficult tasks, where traditional single-classifier approaches may fall short. The ability to accurately assess public sentiment regarding mental health can inform both clinical practices and social interventions. However, there are limitations to consider, such as potential bias within the dataset and challenges posed by informal language used on Twitter. Future research could improve upon this model by integrating more advanced natural language processing techniques, such as transformer based. Additionally, expanding the dataset to include data from other social media platforms such as Facebook or Instagram could further improve the model's robustness and generalizability across various demographics.

REFERENCES

[1] W. H. Organization, “WHO highlights urgent need to transform mental health and mental health care.” [Online]. Available: https://www.who.int/news/item/17-06-2022-who-highlights-urgent-need-to-transform-mental-health-and-mental-health-care

[2] J. A. Naslund, A. Bondre, J. Torous, and K. A. Aschbrenner, “Social Media and Mental Health: Benefits, Risks, and Opportunities for Research and Practice,” *J. Technol. Behav. Sci.*, vol. 5, no. 3, pp. 245–257, 2020, doi: 10.1007/s41347-020-00134-x.

[3] S. Chancellor and M. De Choudhury, “Methods in predictive techniques for mental health status on social media: a critical review,” *npj Digit. Med.*, vol. 3, no. 1, 2020, doi: 10.1038/s41746-020-0233-7.

[4] N. Braig, A. Benz, S. Voth, J. Breitenbach, and R. Buettner, “Machine Learning Techniques for Sentiment Analysis of COVID-19-Related Twitter Data,” *IEEE Access*, vol. 11, pp. 14778–14803, 2023, doi: 10.1109/ACCESS.2023.3242234.

[5] Y. DIng, X. Chen, Q. Fu, and S. Zhong, “A Depression Recognition Method for College Students Using Deep Integrated Support Vector Algorithm,” *IEEE Access*, vol. 8, pp. 75616–75629, 2020, doi: 10.1109/ACCESS.2020.2987523.

[6] N. Al Asad, M. A. Mahmud Pranto, S. Afreen, and M. M. Islam, “Depression Detection by Analyzing Social Media Posts of User,” in *2019 IEEE International Conference on Signal Processing, Information, Communication & Systems (SPICSCON)*, 2019, pp. 13–17. doi: 10.1109/SPICSCON48833.2019.9065101.

[7] Z. N. Vasha, B. Sharma, I. J. Esha, J. Al Nahian, and J. A. Polin, “Depression detection in social media comments data using machine learning algorithms,” *Bull. Electr. Eng. Informatics*, vol. 12, no. 2, pp. 987–996, 2023, doi: 10.11591/eei.v12i2.4182.

[8] A. Renaldi and W. Maharani, “Depression Detection of User in Media Social Twitter Using Random Forest,” *J. Inf. Syst. Res.*, vol. 3, no. 4, pp. 410–416, 2022, doi: 10.47065/josh.v3i4.1837.

[9] R. H. H. Aziz and N. Dimililer, “Twitter Sentiment Analysis using an Ensemble Weighted Majority Vote Classifier,” *3rd Int. Conf. Adv. Sci. Eng. ICOASE 2020*, pp. 103–109, 2020, doi: 10.1109/ICOASE51841.2020.9436590.

[10] W. Bin Tahir, S. Khalid, S. Almutairi, M. Abohashrh, S. A. Memon, and J. Khan, “Depression Detection in Social Media: A Comprehensive Review of Machine Learning and Deep Learning Techniques,” *IEEE Access*, vol. 13, no. December 2024, pp. 12789–12818, 2025, doi: 10.1109/ACCESS.2025.3530862.

[11] K. E. Hoque and H. Aljamaan, “Impact of hyperparameter tuning on machine learning models in stock price forecasting,” *IEEE Access*, vol. 9, pp. 163815–163830, 2021, doi: 10.1109/ACCESS.2021.3134138.

[12] InFamousCoder, “Depression: Twitter Dataset + Feature Extraction.” [Online]. Available: https://www.kaggle.com/datasets/infamouscoder/mental-health-social-media/data

[13] Z. Jianqiang and G. Xiaolin, “Comparison research on text pre-processing methods on twitter sentiment analysis,” *IEEE Access*, vol. 5, no. c, pp. 2870–2879, 2017, doi: 10.1109/ACCESS.2017.2672677.

[14] F. D. Wibowo, I. Palupi, and B. A. Wahyudi, “Image Detection for Common Human Skin Diseases in Indonesia Using CNN and Ensemble Learning Method,” *J. Comput. Syst. Informatics*, vol. 3, no. 4, pp. 527–535, 2022, doi: 10.47065/josyc.v3i4.2151.

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1017−1027
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.7763

[15] B. A. Mustofa, W. Laksito, and Y. Saptomo, "Journal of Artificial Intelligence and Engineering Applications Use of Natural Language Processing in Social Media Text Analysis," Journal of Artificial Intelligence and Engieenering Applications, vol. 4, no. 2, pp. 2808–4519, 2025, [Online]. Available: https://ioinformatic.org/

[16] A. W. Pradana and M. Hayaty, "The Effect of Stemming and Removal of Stopwords on the Accuracy of Sentiment Analysis on Indonesian-language Texts," *Kinet. Game Technol. Inf. Syst. Comput. Network, Comput. Electron. Control*, vol. 4, no. 3, pp. 375–380, 2019, doi: 10.22219/kinetik.v4i4.912.

[17] R. Pramana, Debora, J. J. Subroto, A. A. S. Gunawan, and Anderies, "Systematic Literature Review of Stemming and Lemmatization Performance for Sentence Similarity," *Proc. 2022 IEEE 7th Int. Conf. Inf. Technol. Digit. Appl. ICITDA 2022*, no. November 2022, 2022, doi: 10.1109/ICITDA55840.2022.9971451.

[18] T. E. Ramya and S. Sindhupriya, "An Effective Approach for Mental Health Prediction Using Machine Learning Algorithm," *Int. J. Eng. Res. Technollogy*, vol. 10, no. 13, pp. 81–84, 2022.

[19] wesam ahmed, N. Semary, K. Amin, and M. Adel Hammad, "Sentiment Analysis on Twitter Using Machine Learning Techniques and TF-IDF Feature Extraction: A Comparative Study," *IJCI. Int. J. Comput. Inf.*, vol. 10, no. 3, pp. 52–57, 2023, doi: 10.21608/ijci.2023.236052.1128.

[20] H. Bichri, A. Chergui, and M. Hain, "Investigating the Impact of Train / Test Split Ratio on the Performance of Pre-Trained Models with Custom Datasets," *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 2, pp. 331–339, 2024, doi: 10.14569/IJACSA.2024.0150235.

[21] M. S. Santos, J. P. Soares, P. H. Abreu, H. Araujo, and J. Santos, "Cross-validation for imbalanced datasets: Avoiding overoptimistic and overfitting approaches [Research Frontier]," *IEEE Comput. Intell. Mag.*, vol. 13, no. 4, pp. 59–76, 2018, doi: 10.1109/MCI.2018.2866730.

[22] P. M. Mathapati, A. S. Shahapurkar, and K. D. Hanabaratti, "Sentiment Analysis using Naïve bayes Algorithm," *Int. J. Comput. Sci. Eng.*, vol. 5, no. 7, pp. 75–77, 2017, doi: 10.26438/ijcse/v5i7.7577.

[23] D. Pradana and E. Sugiharti, "Implementation Data Mining with Naive Bayes Classifier Method and Laplace Smoothing to Predict Students Learning Results," *Recursive J. Informatics*, vol. 1, no. 1, pp. 1–8, 2023, doi: 10.15294/rji.v1i1.63964.

[24] J. Nayak, B. Naik, and H. S. Behera, "A Comprehensive Survey on Support Vector Machine in Data Mining Tasks: Applications & Challenges," *Int. J. Database Theory Appl.*, vol. 8, no. 1, pp. 169–186, 2015, doi: 10.14257/ijdta.2015.8.1.18.

[25] D. Mustafa Abdullah and A. Mohsin Abdulazeez, "Machine Learning Applications based on SVM Classification A Review," *Qubahan Acad. J.*, vol. 1, no. 2 SE-Articles, pp. 81–90, Apr. 2021, doi: 10.48161/qaj.v1n2a50.

[26] I. Palupi, B. ari Wahyudi, N. AL Mamuda, and A. Shabrina, "Predicting Forest Fire Hotspots with Carbon Emission Insights Using Random Forest and Gradient Boosting Regression," *Int. J. Inf. Commun. Technol.*, vol. 9, no. 2, pp. 137–149, 2023, doi: 10.21108/ijoict.v9i2.865.

[27] S. L. Setyowati, A. Qalbi, R. Aristawidya, B. Sartono, and A. R. Firdawanti, "Optimizing Random Forest Parameters with Hyperparameter Tuning for Classifying School-Age KIP Eligibility in West Java," *Jambura J. Math.*, vol. 7, no. 1, pp. 40–48, 2025, doi: 10.37905/jjom.v7i1.28736.

[28] M. S. Hashim and A. A. Yassin, "Breast Cancer Prediction Using Soft Voting Classifier Based on Machine Learning Models," *IAENG Int. J. Comput. Sci.*, vol. 50, no. 2, 2023.

[29] Ž. Vujović, "Classification Model Evaluation Metrics," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 6, pp. 599–606, 2021, doi: 10.14569/IJACSA.2021.0120670.