Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1122−1130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.8015

Classification of English Language Anxiety Using Support Vector Machine on Twitter User

Ericho Marozi, Kemas Muslim Lhaksmana*

Fakultas Informatika, Telkom University, Bandung, Indonesia
Email: [1]erichomarozi@student.telkomuniversity.ac.id, [2,*]kemasmuslim@telkomuniversity.ac.id,
Correspondence Author Email: kemasmuslim@telkomuniversity.ac.id
Submitted: **15/07/2025**; Accepted: **01/09/2025**; Published**: 02/09/2025**

Abstract−This study aims to classify expressions of language anxiety in English as a foreign language, as reflected in user-generated texts on Twitter. The research applies machine learning approaches Support Vector Machine (SVM) and Convolutional Neural Network (CNN) to perform automatic classification of anxiety levels. The dataset was collected through Twitter crawling, filtered for relevance, and annotated manually using a three-point scale (low, medium, high) based on psychological indicators such as fear of speaking, avoidance, and self-perceived inability. Preprocessing included text normalization, tokenization, stopword removal, and feature extraction using TF-IDF with unigram to trigram representations. Model training was conducted on a balanced dataset, and performance was evaluated through cross-validation and tuning of key hyperparameters. SVM achieved the highest accuracy of 98.40%, showing strong stability across various test conditions. CNN initially performed competitively but experienced a slight performance drop after tuning, suggesting its sensitivity to parameter settings and data volume. The findings demonstrate that SVM is more robust and suitable for limited-data environments, making it a practical tool for classifying psychological traits like language anxiety in digital communication. This research offers insight into the potential of machine learning in psychological and linguistic analysis, especially through social media platforms.

Keywords: Languange Anxiety; Text Classification; Support Vector Machine; Convolutional Neural Network; TF-IDF

1. INTRODUCTION

The ability to speak foreign languages is a crucial aspect in today's era of globalization, particularly in the fields of education and work, which are now increasingly open across national borders. However, not all individuals can master or use foreign languages fluently [1]. One of the common psychological barriers encountered is foreign language anxiety (language anxiety), which refers to feelings of anxiety or fear that arise when someone must express ideas or communicate, either verbally or in writing, in a foreign language. Such an anxiety not only affects academic performance but also reduces self-confidence as well as motivation for future study [2]. As such, the use of text classification models is significant in determining the number of people language anxious, primarily Twitter users exhibiting features of depression and anxiety when communicating. Such classification techniques for texts are classification methods for data categorising as per agreed features on the basis of attributes shared in common in order to belong to relevant categories [3], [4].

Past works apply tech in language anxiety classification. The current study aims at enhancing knowledge on psychological feelings when a person is learning a foreign language. Shaurya et al.[5]. Conducted a study on the classification of the anxiety levels encountered by engineering students in India through the application of machine learning models, on a basis of questionnaire data consisting of 40 questions on the causes and the effects on anxiety. 127 respondents participated in the questionnaire. Cronbach's alpha and Pearson's correlation test were used in ensuring the study retained reliability and validity in the data. The anxiety levels were classed in three categories, mild, moderate, and severe. The classification was performed in the application of different machine learning algorithms, Naive Bayes, Decision Tree, Random Forest, and Support Vector Machine (SVM). The test outcomes showed that accuracy in the model was between 71% and 79% in which Random Forest offered the maximum at 78.9%. The study, however, encounters various weaknesses, for instance, a small sample size, which is not inclusive in geographical coverage, in which the study covers only an area, making the study not very generalizable. Additionally, classification in an application using a single category in a questionnaire may inhibit further knowledge on factors causing and precipitating anxiety. In another study, Kang et al.[6] found that both Support Vector Machine (SVM) and Naive Bayes were effective for classifying mental stress from ECG signals, but observed that SVM achieved higher accuracy (97.6%) than Naive Bayes, underlining the importance of model selection and parameter tuning in mental health detection tasks. In another study, Hussain et al.[7] developed a mental health chatbot that utilized Recurrent Neural Networks (RNN), specifically designed to deliver cognitive behavioral therapy-based responses. The system aimed to classify and respond to user feedback effectively, achieving high performance across metrics such as accuracy, precision, recall, and F1-score, despite challenges related to dataset size and generalization.

The Support Vector Machine (SVM) algorithm is quite popular in data classification. Kharisma et al. [8], focused their research on testing the effectiveness of the Support Vector Machine (SVM) algorithm in identifying anxiety levels in patients at the Emergency Room of Royal Prima Hospital in Medan. The data used was sourced from observations of patients at the hospital's emergency department, which risks bias as it only represents one location and a limited population. This limits the study's ability to be generalized to a broader context. The SVM algorithm was chosen for its ability to classify medical conditions based on vital data and patient characteristics. However, the limited scope of data from a single institution poses a significant challenge in expanding the application of the research

This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1122−1130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.8015

results. Nevertheless, the findings indicate that the SVM model is capable of accurately identifying patient anxiety levels, supported by consistent vital feature patterns in distinguishing each anxiety category. In another study conducted by Tisha et al.[9], Demographic data, social media interaction patterns, and physiological indicators were utilized to predict and detect anxiety levels early. The information used included age, gender, educational background, place of residence, media usage intensity, well-being scores, and anxiety levels obtained from datasets available on Kaggle. Anxiety levels were grouped into three classes: low, moderate, and high, which were then converted into numerical labels 0, 1, and 2 as classification targets. Research methods comprised preliminary descriptive analysis, data normalization for Z-scores, dimension reduction through PCA, and classification algorithm implementation such as SVM, Decision Tree, Naïve Bayes, and KNN. The performance of the models was assessed against accuracy, precision, recall, and F1-score, and the best-performing model was the SVM, which attained an accuracy of about 95% and surpassed the threshold. Nevertheless, the present study remains limited in that it is dependent on non-normal distribution datasets, and the model is susceptible to overfitting owing to the small sample size. The study further lacks external validation. The research, regardless, indicates that the SVM is a promising technique for effective classification of anxiety levels and can be applicable in effective early detection. The study was further replicated by Tsania et al.[10]. With the aim of exploring the use of the Support Vector Machine (SVM) algorithm in classifying depression, anxiety, and stress levels among Facebook users. Data was collected from active Facebook members who filled in the DASS-21 questionnaire, which was made available on the internet through social media such as Twitter and Facebook using Google Forms. The procedures for the methods comprised questionnaire administration in order to collect data, anonymisation for non-personal data, and transforming and processing data for classification. The findings on test results indicated that the model for SVM proved very high in performance, achieving an accuracy value of 98.96%, precision value of 99.15%, a recall value of 97.26%, and an F1-score value of 95.75%. The study, however, remains limited in that convenience sampling was utilised, and the study can be subject to bias as the data was collected using self-reporting on social media, which can confine the generalizability for the findings to a larger population.

This study makes a contribution to text-based psychological categorisation, but focuses primarily on the language anxiety phenomenon. The research makes use of Support Vector Machine (SVM) algorithms and Natural Language Processing (NLP) methods in order to construct an expansive classification pipeline. The classification pipe is comprised of five core scenarios, such as changes in the proportion between test and train data, the use of n-grams as a feature representation and TF-IDF as a feature extractor, capping the number of features in the TF-IDF vectorizer, the use of cross-validation methods, and hyperparameter tuning in order to improve model performance. Accuracy, precision, recall, and F1-score were used for evaluation. In addition to centring on model performance, the study exhibits the prospective ability of machine learning in automatically identifying language-related anxiety using text analysis. The findings for the study are anticipated to offer an initial foundation for constructing computer-based systems to identify psychological disorders in their early phases, as well as promote the increased utilisation of technology in the mental health and education domains.

2. RESEARCH METHODOLOGY

The system developed in this research is illustrated in Figure 1. The process of system development is comprised of the data labeling and collection process using crawling methods, followed by pre-processing on the text data. The extraction of features is done using the N-gram and TF-IDF methods. The data is further split in order to become the test data and the training data. The classification models adopted include the Convolutional Neural Network (CNN) and Support Vector Machine (SVM) algorithms, which are then tested in order to determine their efficiencies in classifying the levels of anxiety when communicating in a foreign language.

The Figure 1 illustrates the complete process of this research, which begins with the collection of raw data from social media platforms, specifically Twitter, through the crawling method. This technique is used to capture user posts that reflect their experiences in learning or using a foreign language. Given the highly unstructured nature of the data obtained, further processing is required to prepare the data for use in training classification models. A critical stage in this process is data labelling, which is performed manually by human annotators. Each text is labelled to represent the user's anxiety level based on psychological indicators adapted from foreign language anxiety theory. The involvement of annotators with an understanding of linguistic and emotional aspects is crucial to ensure the semantic accuracy of the data, so that the labelling results can serve as a valid reference in the supervised learning model training and testing process. After the labelling process is complete, the data is then processed through a preprocessing stage. This stage involves cleaning the text of irrelevant symbols, punctuation marks, and non-alphabetic characters, followed by tokenisation to separate words within sentences. In addition, common words that do not contribute significantly to the classification process (stopwords) are also removed. The cleaned text data is then feature-extracted using two main approaches: TF-IDF (Term Frequency–Inverse Document Frequency), which assesses the importance of words in a document, and N-gram, which is used to capture contextual relationships between words based on their order of appearance. The features generated from this process are then divided into two subsets, namely training data and test data, with varying division ratios (such as 70:30, 80:20, and 90:10) for experimental purposes. The training data is used to build the classification model, while the test data is used to evaluate the model's performance on new

Copyright © 2025 **Author**, Page 1123
This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1122−1130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.8015

data that has never been seen before. In building the classification model, two primary machine learning algorithms are employed: the Support Vector Machine (SVM) and the Convolutional Neural Network (CNN). SVM is chosen for its ability to manage high-dimensional data representations such as those resulting from TF-IDF. At the same time, CNN, which originates from image processing architecture, is utilised to detect spatial patterns in the matrix representation of text data. To ensure that the model does not overfit the training data and has good generalisation capabilities, cross-validation was performed using different schemes, such as CV-3, CV-5, and CV-10. Additionally, hyperparameter tuning was conducted to optimise model performance, including adjusting the kernel type in SVM and configuring the number of filters and layers in CNN.

The final step in this process is a comprehensive evaluation of the classification system that has been built. The evaluation is conducted using various performance metrics, including accuracy, precision, recall, and the F1-score. The purpose of this evaluation is to measure the effectiveness of the model in detecting and classifying users' anxiety levels based on the text they generate, thereby determining to what extent the model optimally performs its classification task.

Figure 1. Flowchart of Design System

2.1 Anxiety

Physical and emotional stress may cause adverse emotions or feelings, which are typical of anxiety [11]. There are different forms of anxiety, specifically language anxiety, which is experienced when a person goes through some emotional states like fear, nervousness, and distractibility when either acquiring or making use of a second or foreign language. This is a condition that can hamper the process of learning because, in their normal course, students can be hindered from optimizing language skills [12].

2.2 Classification

Classification is a method for obtaining a set of functions or models capable of separating and representing different classes or concepts in a data set. The main task is enabling the model to be used for predicting the class category for objects for which the class is not known [13], [14].

2.3 Dataset

The dataset was created by web crawling using the Tweet Harvest tool [v2.6.1] implemented in Google Colab, from October 2021 to January 2022, targeting Indonesian-language tweets using keywords such as 'frustasi', 'cemas', 'takut', and related terms [15], [16]. This resulted in 5,200 raw tweets. After filtering and preprocessing, a total of 5,604 annotated texts were used. The dataset includes one main entity in a text form; that may be a narrative, opinion, or comment, each observation along with one binary label, which is the level of language anxiety, where 0 indicates the user with no anxiety and 1 indicates the user with language anxiety. The binary labeling was done by three annotators using a rubric based on psychological indicators such as fear of speaking, avoidance behavior, self-perceived incompetence, and negative emotional expressions. Inter-rater agreement was calculated using Fleiss' Kappa (κ = 0.78), indicating substantial agreement. After capturing the data, the web crawled data undergoes data pre-processing (cleaning of irrelevant characters, stopword removal, and text representation using the term frequency-inverse document frequency (TF-IDF) method). The cleaned dataset is used to train and evaluate a classification model based on the Support Vector Machine (SVM) algorithm.

Copyright © 2025 **Author**, Page 1124
This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1122−1130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.8015

2.4 Data preprocessing

Data preprocessing is a crucial first step in data processing, where various text manipulation techniques are applied to prepare the data before it enters the advanced analysis stage. This process includes removing irrelevant text elements, breaking the text into tokens or words, filtering out less valuable words such as stopwords, and applying stemming techniques to return words to their basic form. Each of these processes plays an essential role in ensuring that the data obtained is clean and usable, thus becoming the primary foundation for the success of a study[17],[18]. The data preprocessing stage serves as a crucial first step in preparing text for classification, ensuring it's in the right shape for effective analysis by models like Support Vector Machines (SVM). It starts with case folding, where all text is converted to lowercase to avoid inconsistencies from capitalization. Next, the text is cleaned using regular expressions to strip out anything that's not a letter—like numbers, punctuation, symbols, or links. After that, common filler words such as "the," "and," or "or" are removed to focus on more meaningful content. The refined text is then broken down into tokens, which lays the groundwork for turning it into numerical data. This is done using TF-IDF (Term Frequency–Inverse Document Frequency), a method that assigns weights to words based on how often they appear in a specific document compared to the rest of the dataset, helping the model identify which terms truly stand out. To enrich the context representation, in some scenarios, n-gram approaches (such as unigrams and bigrams) and feature number restrictions (max_features) are also employed. Through this series of steps, the initially unstructured text data is successfully processed into a numerical representation ready for use in classifying language anxiety using the SVM algorithm.

2.5 TF-IDF

TF-IDF weighting comprises two primary components: Term Frequency (TF) and Inverse Document Frequency (IDF). TF evaluates the frequency of a word in a document; the more frequently it occurs, the higher its TF value. Meanwhile, IDF counts the number of documents containing that word in the entire document collection [19]. Consequently, TF-IDF is a technique utilized to assess the importance of a word in a document, considering how often it appears. This weight indicates the significance of the word within the document's content. The more often a word appears, the greater the weight value, which indicates that the word has an important role in understanding the content of the document [20].

2.6 N-Gram

The N-Gram approach is among the most efficient techniques for extracting features from text data, utilized in both data mining and text processing applications. This approach segments strings into substrings of n characters each, creating a thorough depiction of words and characters [21] [22].

2.7 *Support Vector Machine (SVM)*

Support Vector Machine (SVM) is a supervised learning technique utilized for classification and regression activities by identifying patterns within data. This algorithm operates by identifying the optimal hyperplane that can separate data according to its class, acting as a dividing line between categories. During its classification process, SVM takes labeled training data for every class and then creates a model that can categorize new data with an unknown category [23]. A different study by Wahyu et al. explains that the Support Vector Machine (SVM) algorithm in machine learning uses a hypothesis space created by linear functions [24].

2.8 *Convolutional Neural Network (CNN)*

A Deep Learning algorithm created for processing visual information is the Convolutional Neural Network (CNN). This algorithm takes image input and is able to identify crucial features or objects present within it, allowing the system to learn and recognize the image. Due to this ability, CNN can reliably differentiate one image from another [25]. In another study by Fani et al., it is explained that Convolutional Neural Networks (CNNs) are an advanced form of Multilayer Perceptrons (MLPs), which are generally used for data processing. CNNs are classified as Deep Neural Networks due to their layered and complex network structure[26].

3. RESULT AND DISCUSSION

3.1 Data Split Scenario

In this study, tests were conducted on three split ratio variations, namely 90:10, 80:20, and 70:30. Both models were tested, and the Table 1 show results of testing these ratios on both models:

Table 1 Data Split Scenario Result

Split Ratio	SVM Accuracy (%)	CNN Accuracy (%)
90:10	93.63	91.77
80:20	92.70	90.58
70:30	92.83	89.55

This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1122−1130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.8015

Based on Table 1, it can be observed that both SVM and CNN achieve the highest performance at a 90:10 split ratio, specifically 93.63% for SVM and 91.77% for CNN. This is reasonable because the larger the proportion of training data, the greater the chance for the model to learn from patterns in the available data. The SVM model shows stable performance with accuracy ranging from 92.70% to 93.63% across all split variations. Interestingly, SVM performs better at the 70:30 split than at the 80:20 split. This may be due to the characteristics of the test data at this split not being too complex, so even with less training data, it can still produce good predictions. The total performance difference between ratios with the SVM model is minimal, less than 1%. This indicates that the model is quite reliable even with a reduced amount of training data.

In this case, the CNN model exhibits a significantly more pronounced decline in performance as the amount of training data decreases. At a 90:10 ratio, the CNN accuracy is 91.77%. Still, it drops to 90.58% at 80:20 and further to 89.55% at 70:30. This indicates that CNN requires more data and needs a large dataset to achieve optimal performance, as CNN builds feature representations gradually in cases with abundant data through its hierarchical convolutional layers.

From these findings, we can conclude that SVM is more flexible to changes in the split ratio and can still maintain high accuracy even when the amount of training data is reduced. On the other hand, CNN has proven to be more sensitive to lower amounts of training data, with subsequent declines in classification performance. These results reinforce the argument that classical learning algorithms, such as SVM, can provide efficient and effective solutions for text classification, especially when the dataset is not too large. In the context of anxiety detection in English, SVM has proven to be more flexible, responding better to shifts in data distribution, making it a suitable candidate for real-time text-based classification systems in dynamic environments.

3.2 Text Representation Scenario

This study tested five text representations: unigram, bigram, trigram, unigram-bigram combination, and unigram-bigram-trigram combination. Each approach has a different level of granularity in capturing word structure and context. All tests were conducted with a 90:10 data split ratio, which has been previously proven to yield the best accuracy for both models. The accuracy results of the SVM and CNN models for each text representation are presented in Table 2.

Table 2 Text Representation Scenario Result

Representasi Teks	SVM Accuracy (%)	CNN Accuracy (%)
Unigram	93.63	91.77
Bigram	82.22	82.22
Trigram	66.04	67.10
Unigram + Bigram	92.04	91.51
Unigram + Bigram + Trigram	90.45	93.63

According to Table 2, the unigram representation achieved the highest performance using the SVM model, with an accuracy of 93.63%. In other words, the SVM model performed best when the data was represented as single words, disregarding the context of more extended word sequences. This may be because SVM works better with simple feature representations and fewer dimensions, as it is highly susceptible to overfitting when the features are too complex. Conversely, the performance of the SVM model decreases drastically when bigrams and trigrams are used alone, with accuracies of 82.22% and 66.04%, respectively. As the number of n-gram components increases, the dimensions and sparsity that affect the difficulty of learning increase significantly. Additionally, the unigram-bigram combination achieves a relatively high performance of 92.04%, but it is still lower than that of unigrams. The performance decline of the trigram combination reinforces the assumption that the SVM model is not suitable for learning on feature representations with excessive complexity without a significant amount of data.

However, the CNN model reveals a slightly different pattern. Although the unigram representation achieves a relatively high recognition rate of 91.77% accuracy, the highest performance is achieved at the unigram-bigram-trigram recognition level, with an accuracy of 93.63%. This indicates that CNN can utilize a broader and more complex word context compared to the NB model, driven by its architecture that allows for hypotheses about spatial word relationships, which can be made possible by the convolution layer method. However, the use of bigrams or trigrams in isolation also results in a significant decrease in CNN performance accuracy, at 82.22% and 67.10%, respectively. These results show that although CNN can recognize and process more complex word structures, multi-level combination context representation is still very much needed for optimal recognition. Even though the accuracy of the unigram-bigram representation is very close to the maximum value of 91.51%, it is lower than the combination of three different n-gram levels. Ultimately, the results show that the SVM model is best suited for unigram representations. Abstract text representations provide features that are simpler and more independent, thereby enhancing the model's stability while maintaining consistent accuracy levels. Conversely, CNN performance is optimal when using a combination of unigrams, bigrams, and trigrams. This suggests that the CNN model is more stable in capturing complex patterns with more comprehensive word contexts. Therefore, the text representation to be used also needs to be considered in light of the model's characteristics. For classic models, such as SVM, simple

This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1122−1130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.8015

representations tend to be more efficient. However, for deep learning-based models, like CNN, text representations that capture context tend to be more optimal.

3.3 Max Feature Scenario

The testing was conducted using the best configuration for each model. For SVM, a unigram text representation was used, while CNN employed a combination of unigram, bigram, and trigram representations. The ratio of training and testing data was 90:10, following the optimal results from the previous scenario. The accuracy results of both models are shown in Table 3.

Table 3 Max Feature Scenario Result

Max Feature	SVM Accuracy (%)	CNN Accuracy (%)
3.000	94.16	94.16
5.000	93.63	93.63
7.000	92.83	92.04

Table 3 shows that feature restriction has a positive impact on the performance of both models. Both SVM and CNN achieved the highest accuracy of 94.16% when the number of features was limited to 3,000 words. This indicates that the number of features does not always correlate with improved model performance; in fact, it can reduce accuracy if the added features are redundant or uninformative. In SVM, accuracy decreases gradually as the number of features increases. From 94.16% at 3,000 features, it reduces to 93.63% at 5,000, and 92.83% at 7,000 features. This decline can be attributed to the nature of SVM, which is more sensitive to noise and high complexity, where an excessive number of features narrows the separation margin and increases the risk of overfitting. However, a similar decline trend is also observed in the CNN model. The highest accuracy is also achieved at 3,000 features, and gradually decreases at 5,000 and 7,000. Although CNNs have an advantage in processing complex features, this experiment demonstrates that excessive dimensionality still negatively impacts performance. This may occur because most of the additional words have low frequency and minimal contribution to the contextual meaning required by the model. Thus, the results of this scenario show that proper feature selection is a crucial factor in the development of text-based classification systems. Too few features may reduce important information, but too many features can reduce the efficiency and accuracy of the model. The optimal number of features in this experiment was 3,000, which was used for both the SVM and CNN models, and can serve as a starting point for further implementation.

3.4 Cross-Validation Scenario

In this scenario, three cross-validation variations were performed, namely 3-fold (CV-3), 5-fold (CV-5), and 10-fold (CV-10). The accuracy results of both models are shown in Table 4.

Table 4 Cross-Validation Scenario Result

Cross-Validation	SVM Accuracy (%)	CNN Accuracy (%)
CV-3	92.83	92.06
CV-5	93.22	93.60
CV-10	94.26	94.34

Based on Table 4, both the SVM and CNN models show consistent performance improvements as the number of folds in cross-validation increases. This indicates that the models can achieve better generalization when evaluated on a larger number of data subsets. As the number of folds increases, the proportion of the dataset aside for testing in each iteration becomes smaller, providing the model with a broader and more diverse training experience. For SVM, accuracy increased from 92.83% (CV-3) to 94.26% (CV-10). The accuracy of the CNN model showed a similar trend, rising from 92.06% to 94.34% on CV-10. Although the difference in accuracy between CV-5 and CV-10 is not dramatic, it is significant in the context of text classification because, in this case, even minor fluctuations can compromise model performance due to data variations. Applying higher cross-validation, such as CV-10, also provides more accurate performance estimates, making the forecast more stable and less likely to overfit on specific data subsets. This is particularly important for psychological classification tasks in research, such as anxiety detection in spoken language, where data can be highly variable and rich in complex emotional nuances. Additionally, the results show that CNNs are slightly superior to SVMs in the best configuration with CV-10, although the difference is minimal. This suggests that deep learning models, such as CNNs, start to demonstrate their advantages when evaluations are conducted more comprehensively and thoroughly, despite requiring more extensive computational resources.

3.5 Hyperparameter Tuning Scenario

In this scenario, hyperparameter tuning was performed, and the results were compared with those from the cross-validation scenario. The cross-validation scenario was used as a comparison because the results from the cross-validation scenario were greater than those from the Max Feature scenario. The accuracy results of both models are shown in

Copyright © 2025 **Author**, Page 1127
This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1122−1130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.8015

Table 5.

Table 5 Hyperparameter Tuning Scenario Result

Model	Best Scenario (Cross-Validation)	After Hyperparameter Tuning
SVM	94.26%	98.40%
CNN	94.34%	93.34%

Table 5 shows that the hyperparameter setting process greatly influenced the SVM model, increasing the accuracy from 94.26% to 98.40%, with an increase of 4.14%. This increase shows that the optimal combination of parameters is very important in the ability of the SVM model to distinguish data that shows anxiety from those that do not. In comparison, the results of the tuning process on the CNN model showed a decrease in performance, from 94.34% down to 93.34%, with a decrease of 1%. This decrease occurs due to several factors, such as an increase in the complexity of the CNN algorithm, sensitivity to certain parameters, or parameter mismatch to the dataset used. Moreover, the tuning procedure for CNN necessitates a wider investigation of the parameter space and an extended training duration, rendering it more vulnerable to overfitting unless mitigated through strategies such as cross-validation combined with early stopping based on validation loss. These results suggest that SVM is not only more consistent with prior scenario changes but also extremely receptive to parameter adjustments. When configured correctly, the performance of SVM can be greatly enhanced to nearly reach perfection. On the other hand, CNN demands a more careful tuning strategy and frequently requires a substantial amount of data to reach its best performance. In the SVM tuning process, we employed GridSearchCV to explore combinations of key hyperparameters, including the regularization parameter C (0.1, 1, 10), kernel types (linear, rbf, poly), degree (for polynomial kernel, 2 and 3), and gamma (scale, auto). The model was trained using a unigram-based TF-IDF representation with 3,000 features on a balanced dataset. This grid-based approach enabled us to systematically identify the best configuration, which resulted in a significant performance improvement—from 94.26% to 98.40% accuracy. These results confirm the sensitivity of the SVM model to proper hyperparameter tuning, especially in text classification tasks involving subtle psychological labels such as anxiety.

In the CNN tuning process, we explored several hyperparameter combinations manually through grid-like search, focusing on key parameters such as the number of filters (e.g., 16, 32), kernel sizes (e.g., 3, 5), and dropout rates (e.g., 0.3, 0.5). The CNN architecture consisted of a convolutional layer with LeakyReLU activation, followed by max pooling, dropout, a dense hidden layer, and a final sigmoid output layer. The Adam optimizer was used with a fixed learning rate, and binary cross-entropy was the loss function. To reduce overfitting, early stopping based on validation loss was applied in each fold of 10-fold cross-validation. Despite these efforts, the CNN model showed a performance decline after tuning, dropping from 94.34% to 93.34%. This outcome likely stems from the relatively small dataset size, which limits the learning capacity of deeper models. Additionally, no architectural variation (e.g., additional convolutional blocks or transfer learning) was tested. The previously used term "layered validation" referred to this multi-level evaluation process specifically, early stopping for internal validation and k-fold cross-validation for external validation. To avoid confusion, we have replaced it with standard terminology.

3.6 Discussion

To summarize the experimental results, Figure 2 presents a grouped bar chart comparing the accuracy of SVM and CNN across five evaluation scenarios: data split ratios, text representation models, feature counts, cross-validation folds, and hyperparameter tuning. This visualization enables a holistic comparison between the two models under different configurations.

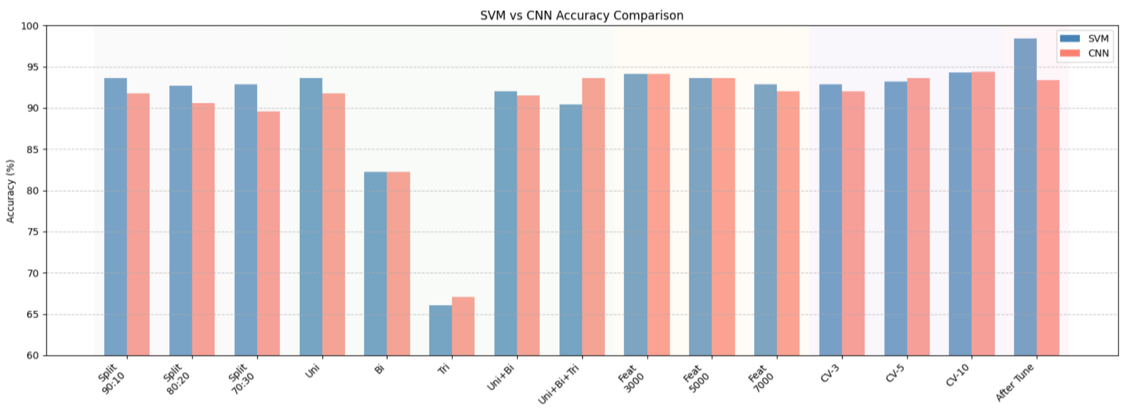

Figure 2 SVM vs CNN Accuracy Comparasion

The chart confirms that SVM consistently achieved higher and more stable accuracy in most scenarios. Notably, SVM performed best with unigram representations and smaller feature sizes, and its accuracy increased

This Journal is licensed under a Creative Commons Attribution 4.0 International License

steadily with more cross-validation folds, peaking at 98.40% after tuning. These findings highlight SVM's robustness and lower sensitivity to parameter variations, making it suitable for tasks involving structured, short-form text like social media posts. In contrast, CNN showed more fluctuating performance, with greater sensitivity to representation and tuning. While CNN performed well on richer feature sets, such as combined n-gram models, its accuracy decreased after tuning—suggesting that its complexity can lead to overfitting when applied to relatively small datasets. CNN benefited from deeper validation folds but remained less consistent overall.

Overall, the visualization strengthens the conclusion that SVM is a reliable and efficient choice for language anxiety classification on modest-sized datasets. Meanwhile, CNN holds potential when supported by richer features and more careful hyperparameter optimization, but requires further tuning and possibly larger data volumes to reach its full effectiveness.

4. CONCLUSION

This study successfully demonstrated that the Support Vector Machine (SVM) algorithm performs exceptionally well in classifying language anxiety based on text data, achieving an accuracy rate of up to 98.40% after hyperparameter tuning. Across five testing scenarios ranging from data split variations, text representation, feature number constraints, cross-validation, to hyperparameter tuning, the SVM model demonstrated greater stability and adaptability compared to the CNN, which, although initially showing competitive performance, tended to experience a decline in accuracy after tuning. The superiority of SVM demonstrates that machine learning-based approaches can be effectively used to process unstructured data, such as social media posts, in detecting symptoms of language anxiety. The classification system developed shows excellent potential to become an automated, fast, and widely applicable early detection tool, particularly in the contexts of education and mental health. However, this study has several limitations, including a relatively small dataset size, a manual data labeling process that is prone to subjective bias, and an unexplored space in CNN architecture. Additionally, the algorithms used are limited to SVM and CNN; therefore, testing other methods, such as ensemble learning or transformer-based models, is necessary in future research to achieve more comprehensive results and stronger generalization.

REFERENCES

[1] A. Deani, G. J. Putri, Adib Alfalah, and A. Perizga, “Kecemasan Penggunaan Bahasa Asing di Kelas AFL Dewasa,” *Jurnal Pengabdian Masyarakat dan Riset Pendidikan*, vol. 3, no. 4, pp. 1594–1603, May 2025, doi: 10.31004/jerkin.v3i4.696.

[2] J. M. Dewaele, E. Botes, and R. Meftah, “A Three-Body Problem: The effects of foreign language anxiety, enjoyment, and boredom on academic achievement,” *Annu Rev Appl Linguist*, vol. 43, pp. 7–22, 2023, doi: 10.1017/S0267190523000016.

[3] K. Rahayu, V. Fitria, D. Septhya, R. Rahmaddeni, and L. Efrizoni, “Klasifikasi Teks untuk Mendeteksi Depresi dan Kecemasan pada Pengguna Twitter Berbasis Machine Learning,” *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 3, no. 2, pp. 108–114, 2023, doi: 10.57152/malcom.v3i2.780.

[4] H. Wang, J. He, X. Zhang, and S. Liu, “A Short Text Classification Method Based on NGram and CNN,” *Chinese Journal of Electronics*, vol. 29, no. 2, pp. 248–254, Mar. 2020, doi: 10.1049/cje.2020.01.001.

[5] S. Bhatnagar, J. Agarwal, and O. R. Sharma, “Detection and classification of anxiety in university students through the application of machine learning,” *Procedia Comput Sci*, vol. 218, no. 2022, pp. 1542–1550, 2022, doi: 10.1016/j.procs.2023.01.132.

[6] M. Kang, S. Shin, G. Zhang, J. Jung, and Y. T. Kim, “Mental Stress Classification Based on a Support Vector Machine and Naive Bayes Using Electrocardiogram Signals,” *Sensors*, vol. 21, no. 23, p. 7916, Nov. 2021, doi: 10.3390/s21237916.

[7] P. Rathnayaka, N. Mills, D. Burnett, D. De Silva, D. Alahakoon, and R. Gray, “A Mental Health Chatbot with Cognitive Skills for Personalised Behavioural Activation and Remote Health Monitoring,” *Sensors*, vol. 22, no. 10, p. 3653, May 2022, doi: 10.3390/s22103653.

[8] K. G. Ginting, N. Prasetyo, A. V. Gunawan, M. Sihombing, and A. A. Nababan, “Classification of Anxiety Levels of IGD Patients at RSU Royal Prima Medan Using Support Vector Machine (SVM) Algorithm,” *Teknika*, vol. 14, no. 2, pp. 255–263, Jul. 2025, doi: 10.34148/teknika.v14i2.1243.

[9] T. Sadariya and S. Verma, “Early Prediction and Detection of Anxiety Level Using Support Vector Machine,” *Lecture Notes in Networks and Systems*, vol. 787 LNNS, no. November, pp. 279–291, 2023, doi: 10.1007/978-981-99-6550-2_22.

[10] T. M. Wijiasih, R. N. S. Amriza, and D. A. Prabowo, “The Classification of Anxiety, Depression, and Stress on Facebook Users Using the Support Vector Machine,” *JISA(Jurnal Informatika dan Sains)*, vol. 5, no. 1, pp. 75–79, 2022, doi: 10.31326/jisa.v5i1.1273.

[11] D. Rhamadian, “Dampak Kecemasan (Anxiety) Dalam Olahraga Terhadap Atlet,” *Journal of Sport Science and Tourism Activity (JOSITA)*, vol. 1, no. 1, p. 31, 2022, doi: 10.52742/josita.v1i1.15433.

[12] S. Han, Y. Li, and S. A. Haider, “Impact of Foreign Language Classroom Anxiety on Higher Education Students Academic Success: Mediating Role of Emotional Intelligence and Moderating Influence of Classroom Environment,” *Front Psychol*, vol. 13, Jul. 2022, doi: 10.3389/fpsyg.2022.945062.

[13] A. H. Nasrullah, “Implementasi Algoritma Decision Tree Untuk Klasifikasi Produk Laris,” *Jurnal Ilmiah Ilmu Komputer*, vol. 7, no. 2, pp. 45–51, 2021, doi: 10.35329/jiik.v7i2.203.

[14] N. B. Putri and A. W. Wijayanto, “Analisis Komparasi Algoritma Klasifikasi Data Mining Dalam Klasifikasi Website Phishing,” *Komputika : Jurnal Sistem Komputer*, vol. 11, no. 1, pp. 59–66, 2022, doi: 10.34010/komputika.v11i1.4350.

[15] L. Rook, M. C. Mazza, I. Lefter, and F. Brazier, “Toward Linguistic Recognition of Generalized Anxiety Disorder,” *Front Digit Health*, vol. 4, Apr. 2022, doi: 10.3389/fdgth.2022.779039.

This Journal is licensed under a Creative Commons Attribution 4.0 International License

Building of Informatics, Technology and Science (BITS)
Volume 7, No 2, September 2025 Page: 1122−1130
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v7i2.8015

[16] J. Shen *et al.*, "Establishment and psychometric characteristics of emotional words list for suicidal risk assessment in speech emotion recognition," *Front Psychiatry*, vol. 13, Nov. 2022, doi: 10.3389/fpsyt.2022.1022036.
[17] A. Nilla and E. B. Setiawan, "Film Recommendation System Using Content-Based Filtering and the Convolutional Neural Network (CNN) Classification Methods," *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, vol. 10, no. 1, p. 17, 2024, doi: 10.26555/jiteki.v9i4.28113.
[18] A. Loiz and Z. K. Abdurahman Baizal, "KNN-Based Music Recommender System with Feedforward Neural Network," *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)*, vol. 10, no. 4, pp. 992–1003, 2024, doi: 10.26555/jiteki.v10i4.30526.
[19] L. Mayasari and D. Indarti, "Klasifikasi Topik Tweet Mengenai Covid Menggunakan Metode Multinomial Naïve Bayes Dengan Pembobotan Tf-Idf," *Jurnal Ilmiah Informatika Komputer*, vol. 27, no. 1, pp. 43–53, 2022, doi: 10.35760/ik.2022.v27i1.6184.
[20] R. A. Supono and M. A. Suprayogi, "Perbandingan Metode TF-ABS dan TF-IDF Pada Klasifikasi Teks Helpdesk Menggunakan K-Nearest Neighbor," *Jurnal RESTI*, vol. 5, no. 5, pp. 911–918, 2021, doi: 10.29207/resti.v5i5.3403.
[21] E. Mulyani, F. P. B. Muhamad, and K. A. Cahyanto, "Pengaruh N-Gram terhadap Klasifikasi Buku menggunakan Ekstraksi dan Seleksi Fitur pada Multinomial Naïve Bayes," *Jurnal Media Informatika Budidarma*, vol. 5, no. 1, p. 264, 2021, doi: 10.30865/mib.v5i1.2672.
[22] W. Li and N. Aletras, "Improving Graph-Based Text Representations with Character and Word Level N-grams," in *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2022, pp. 228–233. doi: 10.18653/v1/2022.aacl-short.29.
[23] A. Noviriandini, H. Hermanto, and Y. Yudhistira, "Klasifikasi Support Vector Machine Berbasis Particle Swarm Optimization Untuk Analisa Sentimen Pengguna Aplikasi Pedulilindungi," *JIKA (Jurnal Informatika)*, vol. 6, no. 1, p. 50, 2022, doi: 10.31000/jika.v6i1.5681.
[24] W. Eko Saputro, H. Yuana, and W. Dwi Puspitasari, "Analisis Sentimen Pengguna Dompet Digital Dana Pada Kolom Komentar Google Play Store Dengan Metode Klasifikasi Support Vector Machine," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 2, pp. 1151–1156, 2023, doi: 10.36040/jati.v7i2.6842.
[25] D. Iswantoro and D. Handayani UN, "Klasifikasi Penyakit Tanaman Jagung Menggunakan Metode Convolutional Neural Network (CNN)," *Jurnal Ilmiah Universitas Batanghari Jambi*, vol. 22, no. 2, p. 900, 2022, doi: 10.33087/jiubj.v22i2.2065.
[26] Verdy and E. Hartati, "Klasifikasi Penyakit Mata Menggunakan Convolutional Neural Network Model Resnet-50," *Jurnal Rekayasa Sistem Informasi dan Teknologi*, vol. 1, no. 3, pp. 199–206, 2024, doi: 10.59407/jrsit.v1i3.529.

This Journal is licensed under a Creative Commons Attribution 4.0 International License