Comparative Analysis of Machine Learning, Deep Learning, and Transformer Models for Multi-Class Emotion Detection on Social Media

Dwi Ngafifudin[1,*], Isnaeni Kustiningsih[2]

[1,2]Magister of Computer Sciences, Amikom Purwokerto University, Indonesia.

ABSTRACT

Emotion detection on social media has become increasingly important for understanding public opinion, behavior, and psychological states. This study presents a comparative analysis of machine learning, deep learning, and transformer-based models for multi-class emotion classification. The models evaluated include Naïve Bayes, Support Vector Machine (SVM), Long Short-Term Memory (LSTM), and Bidirectional Encoder Representations from Transformers (BERT). The dataset was preprocessed through text cleaning, normalization, and label grouping into four classes: positive, negative, neutral, and other. Experimental results show that the LSTM model achieves the highest accuracy of 80.95%, outperforming SVM (79.59%), BERT (77.55%), and Naïve Bayes (73.47%). The findings indicate that deep learning models are more effective in capturing contextual and sequential information compared to traditional approaches. However, the performance of the transformer model is affected by dataset limitations, including small size and class imbalance. This study highlights that model performance is highly dependent on data characteristics and demonstrates that LSTM provides a robust solution for emotion detection in imbalanced social media datasets. Future work may focus on improving performance through data balancing techniques and advanced fine-tuning of transformer models.

Keywords Emotion Detection, Social Media Analysis, Machine Learning, Deep Learning, Transformer Models

INTRODUCTION

The rapid growth of social media platforms has transformed the way individuals express opinions, emotions, and experiences in daily life. Platforms such as Twitter, Facebook, and Instagram generate massive volumes of textual data, reflecting users' emotional states in real time. Emotion detection from social media text has therefore become an important task in natural language processing, with applications in public opinion analysis, mental health monitoring, and decision-making systems [1]. However, analyzing emotions in social media data remains challenging due to the informal nature of the language, including slang, abbreviations, and inconsistent grammatical structures [2].

To address these challenges, various approaches have been proposed, ranging from traditional machine learning methods to advanced deep learning and transformer-based models. Machine learning algorithms such as Naïve Bayes and Support Vector Machine (SVM) have been widely used due to their simplicity and efficiency in handling text classification tasks [3]. These models typically rely on feature extraction techniques such as Bag-of-Words and TF-

Submitted 6 July 2026
Accepted 5 August 2026
Published 1 September 2026

Corresponding author
Dwi Ngafifudin,
25MA41D021@students.amikompurwokerto.ac.id

Additional Information and Declarations can be found on page 237

How to cite this article: D. Ngafifudin and I. Kustiningsih, "Comparative Analysis of Machine Learning, Deep Learning, and Transformer Models for Multi-Class Emotion Detection on Social Media," *J. Digit. Soc.*, vol. 2, no. 3, pp. 225-239, 2026.

IDF representations. Subsequently, deep learning models, particularly Long Short-Term Memory (LSTM), have been introduced to better capture sequential dependencies and contextual relationships within text data [4]. More recently, transformer-based models such as BERT have demonstrated significant improvements by leveraging contextual embeddings and pre-trained language representations, enabling more accurate understanding of semantic meaning in text [5].

Despite these advancements, several limitations remain in existing studies. Many previous works focus on binary sentiment classification rather than multi-class emotion detection, limiting their ability to capture the complexity of human emotions. In addition, most studies report superior performance of transformer-based models; however, these results are often achieved using large-scale and well-balanced datasets. In real-world scenarios, especially in social media data, datasets are frequently small, noisy, and imbalanced, which can significantly affect model performance. Furthermore, comparative studies that simultaneously evaluate machine learning, deep learning, and transformer-based approaches within the same experimental framework are still limited.

Based on these gaps, this study aims to conduct a comprehensive comparative analysis of machine learning, deep learning, and transformer-based models for multi-class emotion detection on social media data. The models evaluated in this study include Naïve Bayes, SVM, LSTM, and BERT. This research contributes by providing empirical insights into how different model architectures perform under real-world conditions, particularly in the presence of imbalanced data. The findings are expected to help researchers and practitioners select appropriate models for emotion detection tasks based on dataset characteristics and practical constraints.

Literature Review and Related Works

Emotion detection from social media text has been widely studied as an important task in natural language processing, particularly for understanding human behavior and public opinion. Early approaches primarily relied on traditional machine learning algorithms such as Naïve Bayes and Support Vector Machine (SVM), which utilize feature extraction techniques such as Bag-of-Words and TF-IDF. These methods are computationally efficient and perform reasonably well in text classification tasks; however, they often struggle to capture contextual and semantic relationships within text data [6], [7], [8].

To overcome these limitations, deep learning approaches have been introduced, including models such as Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), and their variants. These models are capable of capturing sequential dependencies and contextual information in text, leading to improved performance in emotion classification tasks. Several studies have demonstrated that deep learning models outperform traditional machine learning approaches, particularly in handling informal and unstructured social media data [9], [10], [11].

More recently, transformer-based models such as BERT have significantly advanced the field of natural language processing by leveraging contextual embeddings and pre-trained language representations. These models are designed to understand bidirectional context, enabling more accurate

interpretation of semantic meaning in text. As a result, transformer-based models often achieve state-of-the-art performance in various text classification tasks, including sentiment analysis and emotion detection [12], [13], [14].

Despite the strong performance of transformer models, their effectiveness is highly dependent on the availability of large-scale and well-balanced datasets. In practical scenarios, social media datasets are often noisy, imbalanced, and limited in size, which can negatively impact model performance. Some studies report that deep learning models such as LSTM can outperform transformer-based models in cases where the dataset is small or imbalanced, due to their relatively simpler architecture and lower dependency on large training data [15], [16].

Additionally, several comparative studies have been conducted to evaluate the performance of different models in emotion classification tasks. These studies highlight that SVM generally performs better than Naïve Bayes among traditional methods, while deep learning models tend to achieve higher accuracy overall. However, the performance differences between models may vary depending on dataset characteristics, preprocessing techniques, and evaluation metrics [17], [18], [19].

Furthermore, recent research has emphasized the importance of addressing class imbalance in emotion detection tasks. Imbalanced datasets can lead to biased predictions toward dominant classes, reducing the model's ability to accurately classify minority classes. Various techniques, such as resampling and data augmentation, have been proposed to mitigate this issue, although challenges remain in achieving balanced and robust performance across all classes [20].

Based on the existing literature, it is evident that while significant progress has been made in emotion detection, there is still a need for comprehensive comparative studies that evaluate machine learning, deep learning, and transformer-based models under real-world conditions. This study aims to address this gap by providing a systematic comparison of these approaches using an imbalanced social media dataset.

Methodology

This study proposes a comparative framework to evaluate machine learning, deep learning, and transformer-based models for multi-class emotion detection on social media text. The overall research workflow is illustrated in figure 1, which consists of data collection, preprocessing, label mapping, feature extraction, model training, and performance evaluation. This structured pipeline ensures a systematic comparison across different modeling approaches.

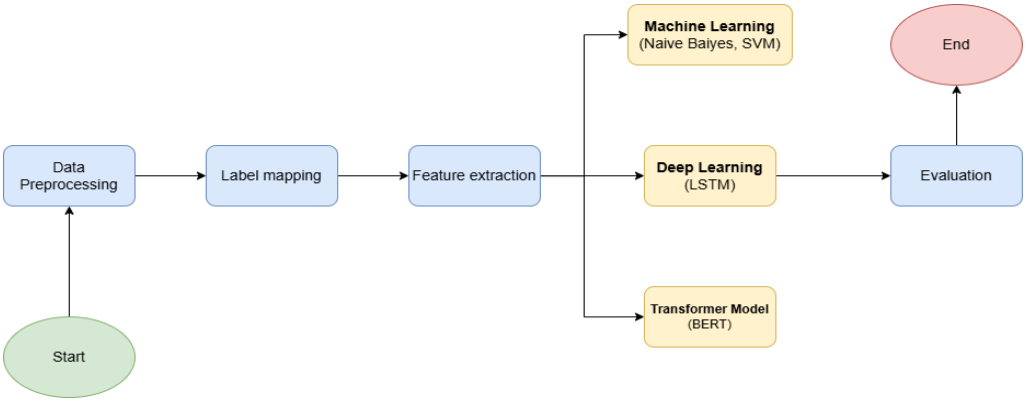

Figure 1 **Research methodology workflow**

Data Collection and Preparation

The dataset used in this study consists of textual data paired with corresponding emotion labels, where each entry represents a piece of text annotated with a specific sentiment category. The primary columns utilized are Text and Sentiment, which are carefully selected from the original dataset to focus on relevant information for analysis. To ensure consistency and improve data quality, several preprocessing steps are applied, including converting all text to a uniform format such as lowercase, removing unnecessary characters like punctuation or special symbols, and handling missing or duplicate entries. Additionally, the sentiment labels are standardized into a consistent set of categories to avoid ambiguity and ensure compatibility with downstream modeling processes. This preparation step is essential for enabling accurate feature extraction and reliable performance in subsequent machine learning or natural language processing tasks.

Text Preprocessing

Text preprocessing is performed to remove noise and improve the overall quality of the dataset before analysis. This process begins by converting all text into lowercase to ensure uniformity and avoid duplication caused by case differences. Next, URLs are identified and removed since they do not contribute meaningful information to sentiment analysis. Punctuation marks and special characters are then cleaned from the text to simplify the structure and focus on the core words. Finally, common stopwords such as "and", "the", and "is" are removed to reduce redundancy and highlight more informative terms, allowing the model to better capture meaningful patterns in the data.

Emotion Label Mapping

The original emotion labels are grouped into four main categories: positive, negative, neutral, and other, in order to reduce complexity and create a more structured classification framework. This grouping helps address issues such as class imbalance and label sparsity by consolidating similar emotions into broader categories. For example, emotions like happiness and excitement may be mapped to positive, while sadness and anger are grouped under negative. The neutral category captures texts with no strong emotional polarity, and the other category accounts for labels that do not clearly fit into the main groups. By simplifying the label space, this approach enhances model generalization,

reduces overfitting, and improves the overall performance and interpretability of the classification model.

Data Filtering and Encoding

Classes with fewer than five samples are removed from the dataset to reduce noise and prevent the model from learning patterns from insufficient or unrepresentative data. Rare classes can negatively impact model performance by introducing bias and instability during training, especially in classification tasks with limited data. After filtering these low-frequency classes, the remaining labels are transformed into numerical form using label encoding, where each category is assigned a unique integer value. This step is necessary to convert categorical labels into a format that can be processed by machine learning algorithms, while maintaining a consistent mapping between the original labels and their encoded representations.

Data Splitting

The dataset is divided into training and testing sets using an 80:20 ratio to ensure that the majority of the data is used for model learning while a separate portion is reserved for unbiased evaluation. Stratified sampling is applied during the split to preserve the original class distribution in both subsets, which is especially important when dealing with imbalanced data. By maintaining proportional representation of each class in the training and testing sets, this approach helps the model learn more effectively and ensures that performance metrics reflect realistic behavior across all categories.

Feature Extraction (TF-IDF)

For machine learning models, text is transformed using TF-IDF representation. The TF-IDF weight is defined as:

$$TF - IDF(t, d) = TF(t, d) \times log\left(\frac{N}{DF(t)}\right) \quad (1)$$

$TF(t, d)$is the frequency of term $t$in document d, $DF(t)$is the number of documents containing term t, and $N$is the total number of documents.

Machine Learning Models

Naïve Bayes is a probabilistic classification algorithm based on Bayes' theorem, which is commonly used in text classification due to its simplicity and efficiency. It calculates the probability of a class given a document by combining prior probabilities of the class and the likelihood of observing the document under that class. The model assumes that all features, such as words in a text, are conditionally independent given the class label, which simplifies computation even though this assumption may not always hold in real-world data. Despite this limitation, Naïve Bayes performs well in many natural language processing tasks because it can handle high-dimensional data and small training samples effectively. The mathematical formulation is given as:

$$P(c \mid d) = \frac{P(d \mid c)\, P(c)}{P(d)} \quad (2)$$

Support Vector Machine is a supervised learning algorithm that is widely used for classification tasks by finding the optimal decision boundary that separates

different classes. The goal of SVM is to construct a hyperplane in a high-dimensional space that maximizes the margin between classes, meaning the distance between the closest data points of each class and the hyperplane is as large as possible. This approach improves generalization and reduces the risk of overfitting. SVM can also use kernel functions to handle non-linear relationships by transforming data into higher-dimensional spaces where separation becomes easier. The decision function of SVM is defined as:

$$f(x) = w \cdot x + b \tag{3}$$

Deep Learning Model (LSTM)

Long Short-Term Memory is a type of recurrent neural network designed to handle sequential data by capturing long-term dependencies and contextual information over time. Unlike traditional recurrent neural networks, LSTM addresses the vanishing gradient problem through the use of memory cells and gating mechanisms, including input, forget, and output gates, which regulate the flow of information. This allows the model to retain important information from earlier time steps while discarding irrelevant data, making it particularly effective for tasks such as text classification, sentiment analysis, and time-series prediction. The core operation of LSTM can be represented as:

$$h_t = LSTM(x_t, h_{t-1}) \tag{4}$$

Transformer Model (BERT)

BERT is a deep learning model based on the Transformer architecture that is designed to understand the contextual relationships between words in a sequence by processing them bidirectionally. Unlike traditional models that read text in a single direction, BERT considers both left and right context simultaneously, allowing it to capture richer semantic meaning. Its core mechanism is the attention function, which determines how much focus should be given to different words in a sentence when encoding a particular word. This enables the model to learn complex language patterns and dependencies, making it highly effective for tasks such as sentiment analysis, question answering, and text classification. The attention mechanism used in BERT is defined as:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{5}$$

Output Layer (Softmax Function)

The softmax function is commonly used in multi-class classification problems to convert raw output scores from a model into probabilities that sum to one. Each output value represents the likelihood of a given input belonging to a specific class, allowing the model to make a final prediction based on the highest probability. The function works by exponentiating each score and normalizing it by the sum of all exponentiated scores, which ensures that all resulting values are positive and comparable across classes. This property makes softmax particularly suitable for classification tasks involving more than two classes, as

it provides a clear probabilistic interpretation of the model's predictions. The mathematical formulation of the softmax function is given as:

$$P(y_i) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}} \quad (6)$$

Performance Evaluation

Model performance is evaluated using accuracy, which measures the proportion of correctly predicted instances out of the total number of predictions. It considers both correctly predicted positive cases and correctly predicted negative cases, making it a straightforward metric for assessing overall model effectiveness. However, accuracy alone may not be sufficient when dealing with imbalanced datasets, as it does not reflect the distribution of errors across classes. The mathematical formulation of accuracy is given as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

In addition to accuracy, confusion matrices are used to provide a more detailed analysis of classification performance by showing the number of true positives, true negatives, false positives, and false negatives for each class. This allows for better understanding of how the model performs across different categories and helps identify specific areas where misclassification occurs.

Algorithm 1: Multi-Class Emotion Detection Pipeline

Input: dataset $D = \{(t_i, y_i)\}_{i=1}^{N}$
Output: predicted labels $\hat{y}$, accuracy
Process:
Start
Text Preprocessing
def clean_text(t_i):
    $t_i \leftarrow lowercase(t_i)$
    $t_i \leftarrow remove\_URL(t_i)$
    $t_i \leftarrow remove\_punctuation(t_i)$
    $t_i \leftarrow remove\_stopwords(t_i)$
    return t_i'
Label Mapping
def map_label(y_i):
    if $y_i \in positive \rightarrow$ return 0
    elif $y_i \in negative \rightarrow$ return 1
    elif $y_i = neutral \rightarrow$ return 2
    else → return 3
Train-Test Split

$$D_{train}, D_{test} = split(D, 0.8)$$

Feature Extraction (TF-IDF)

$$X = TFIDF(t_i')TFIDF(t, d) = TF(t, d) \cdot log(\frac{N}{DF(t)})$$

Naïve Bayes Model
def NB(X):
    $\hat{y} = arg\,max_c\, P(c \mid d)$
    return $\hat{y}_{NB}$
Support Vector Machine
def SVM (X):

$$f(x) = w \cdot x + b$$

    return $\hat{y}_{SVM}$

LSTM Model
def LSTM (X_{seq}):

$$h_t = LSTM(x_t, h_{t-1})\hat{y} = Softmax(Wh_t + b)$$

return $\hat{y}_{LSTM}$
BERT Model
def BERT (X_{bert}):

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V\hat{y} = Softmax(z)$$

return $\hat{y}_{BERT}$

Evaluation

$$Accuracy = \frac{1}{M}\sum_{i=1}^{M} 1(\hat{y}_i = y_i)$$

Model Comparison

$$Result = \{NB, SVM, LSTM, BERT\} BestModel = arg\,max(Result)$$

End

Results

Dataset Distribution

The distribution of emotion labels after preprocessing is illustrated in figure 2, revealing a significant class imbalance within the dataset. The “other” category constitutes the majority of the data, indicating that a large portion of the text does not fall clearly into standard sentiment categories. The positive class represents the second largest portion, while the negative and neutral classes contain substantially fewer samples in comparison. This uneven distribution can impact the learning process of classification models, as they may become biased toward predicting the majority classes more frequently while underperforming on minority classes. As a result, the model may achieve high overall accuracy but fail to correctly identify less represented emotions, highlighting the need for careful evaluation and potential handling of class imbalance during model development.

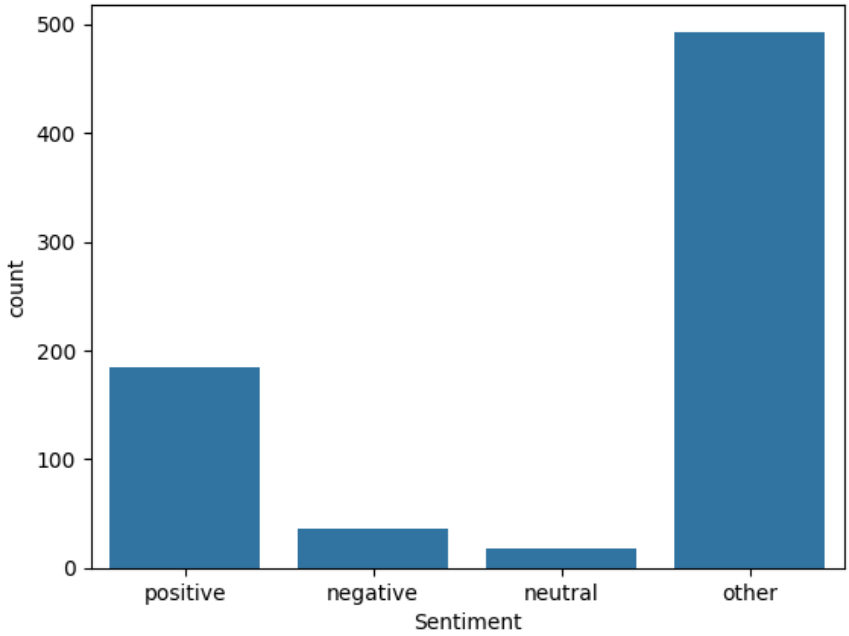

Figure 2 **Emotion distribution after preprocessing**

Model Performance Comparison

The performance of all models is evaluated using accuracy, as presented in table 1, providing a comparative overview of how well each model classifies the emotion labels. The results show that the LSTM model achieves the highest accuracy at 80.95%, indicating its strong capability in capturing sequential patterns and contextual dependencies within the text data. This is followed by the Support Vector Machine with an accuracy of 79.59%, demonstrating competitive performance despite being a traditional machine learning method. The BERT model attains an accuracy of 77.55%, which, while slightly lower, still reflects its effectiveness in understanding contextual relationships through attention mechanisms. Meanwhile, the Naïve Bayes model records the lowest accuracy at 73.47%, likely due to its simplifying assumption of feature independence, which may not fully capture the complexity of natural language. Overall, these results suggest that models capable of handling contextual and sequential information tend to perform better on this task.

Table 1 **Model performance comparison**

Model	Accuracy
Naïve Bayes	0.7347
SVM	0.7959
LSTM	0.8095
BERT	0.7755

Analysis of Machine Learning Models

Naïve Bayes achieved the lowest performance among the evaluated models, indicating its limitation in capturing complex semantic relationships within text data due to its assumption of feature independence. This assumption restricts its ability to model the contextual dependencies between words, which are often crucial in understanding sentiment and emotion. In contrast, the Support Vector Machine demonstrates better performance, as it is well suited for handling high-dimensional feature spaces such as those produced by TF-IDF representations. By finding an optimal decision boundary that maximizes the margin between classes, SVM can effectively distinguish between different categories even when the data is sparse and complex, resulting in improved classification accuracy compared to simpler probabilistic methods.

Analysis of Deep Learning Model (LSTM)

The LSTM model achieves the best performance among all evaluated models, indicating its strong capability in capturing both contextual and sequential information within textual data. By leveraging memory cells and gating mechanisms, LSTM can retain important information from earlier parts of a sequence, which is essential for understanding the meaning and sentiment of text. The confusion matrix of the LSTM model, presented in figure 3, further illustrates its classification behavior by showing the distribution of correct and incorrect predictions across each class. The model performs well on dominant classes, particularly the “other” and positive categories, where a large number

of instances are correctly classified. However, it struggles with minority classes such as neutral and negative, where misclassification occurs more frequently. This suggests that the class imbalance in the dataset affects the model's ability to learn sufficient patterns for underrepresented classes, leading to reduced performance in those categories despite strong overall accuracy.

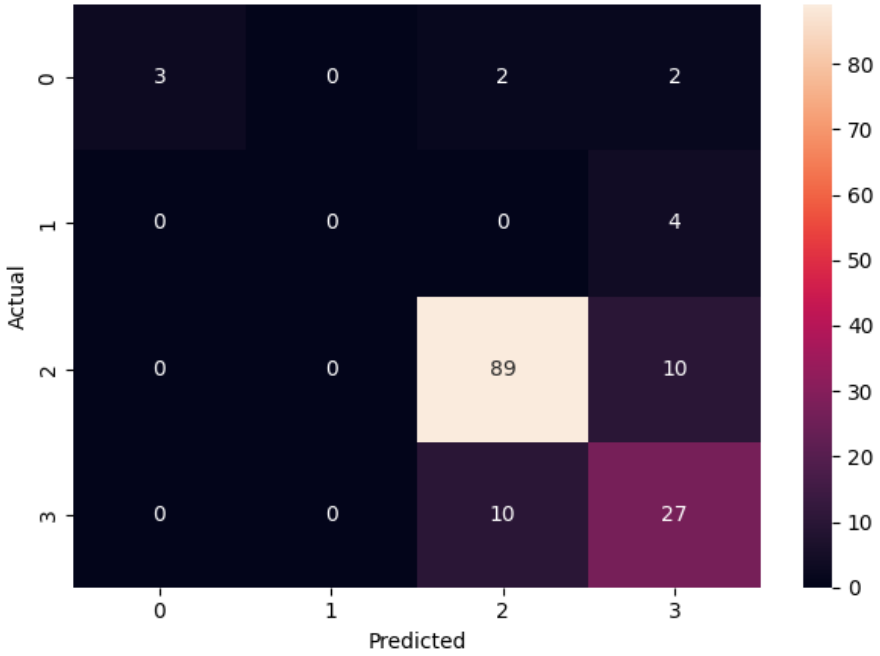

Figure 3 Confusion matrix of LSTM model

Analysis of Transformer Model (BERT)

The BERT model achieves an accuracy of 77.55%, which is lower than that of the LSTM model despite its advanced architecture for capturing deep contextual relationships. While BERT is designed to understand bidirectional context and complex semantic patterns through attention mechanisms, its performance in this study may be influenced by several practical limitations. These include the relatively small size of the dataset, which may not be sufficient to fully leverage BERT's capacity, as well as the presence of class imbalance that can bias predictions toward majority classes. Additionally, a relatively short training duration or limited fine-tuning may prevent the model from converging to optimal performance. As a result, although BERT is theoretically powerful, its effectiveness in this case is constrained by data and training conditions.

Visualization Results

To further analyze model behavior, additional visualizations are presented to provide a clearer understanding of the comparative performance of each model. Figure 4 illustrates the comparison of model accuracy, highlighting the differences in predictive performance across the evaluated methods. The results show that the LSTM model achieves the highest accuracy among all models, reinforcing its effectiveness in handling sequential and contextual information in text data. In comparison, SVM, BERT, and Naïve Bayes exhibit lower accuracy levels, with Naïve Bayes performing the weakest. This visualization supports the

overall findings by clearly demonstrating the superiority of LSTM in this study while also emphasizing the performance gap between deep learning approaches and traditional machine learning models.

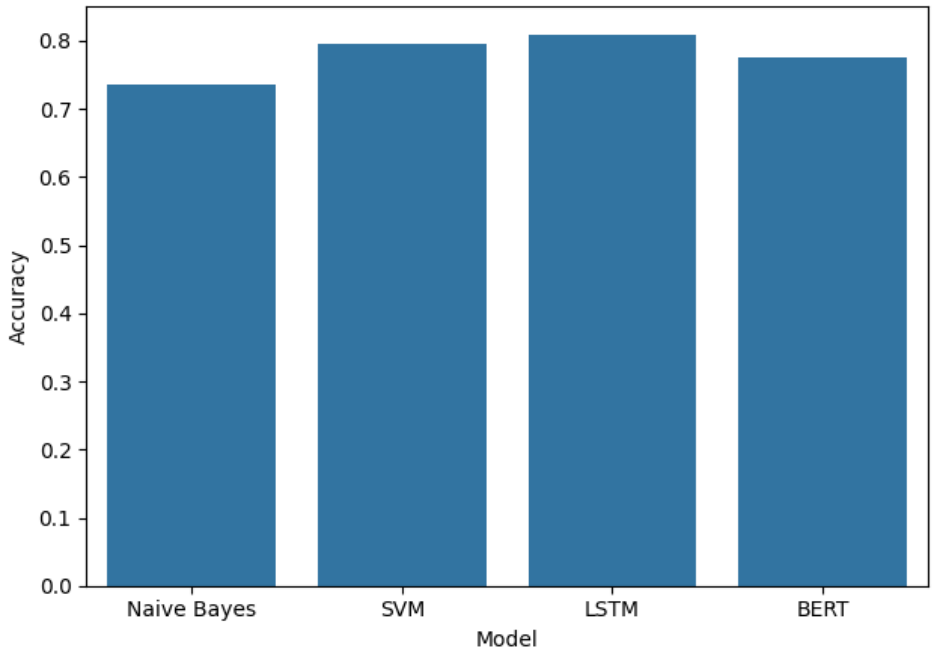

Figure 4 Model performance comparison

Figure 5 illustrates the distribution of text length across different emotion classes, providing insight into how input characteristics vary within the dataset. It can be observed that each emotion category exhibits a distinct range and spread of text lengths, indicating that some classes tend to contain longer or more detailed expressions, while others are represented by shorter and more concise texts. This variation in text length may influence model learning, as models such as LSTM and BERT rely on sequential and contextual information that can be affected by input size. Longer texts may provide richer contextual cues that improve classification, whereas shorter texts may lack sufficient information, leading to higher chances of misclassification. Therefore, understanding this distribution is important for interpreting model performance and identifying potential challenges in handling different types of textual inputs.

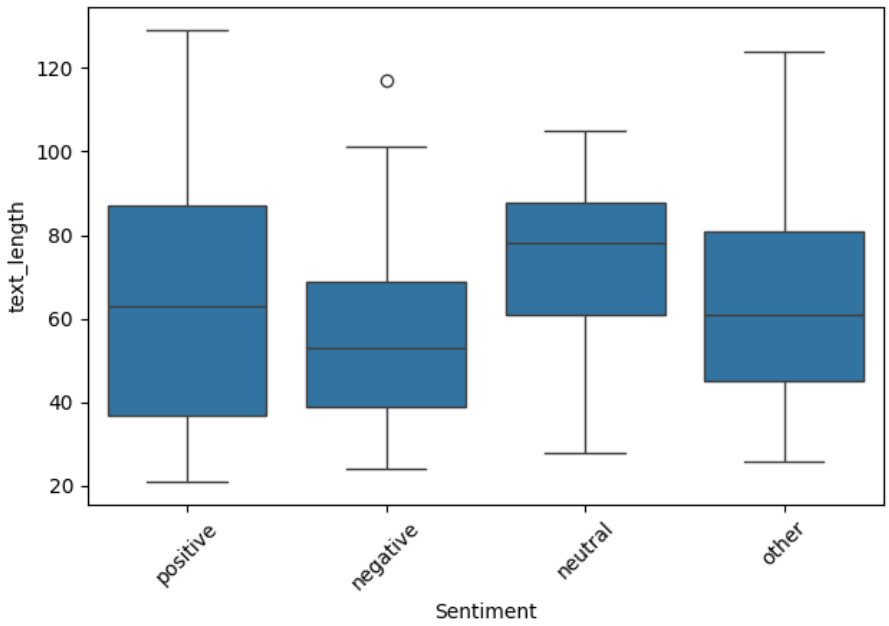

Figure 5 **Text length distribution per emotion**

Discussion

The experimental results demonstrate that the LSTM model outperforms both traditional machine learning and transformer-based approaches in this study. This indicates that sequential deep learning architectures are particularly effective in capturing contextual dependencies within short social media texts. Unlike traditional models such as Naïve Bayes and SVM, which rely heavily on feature engineering techniques like TF-IDF, LSTM is able to learn semantic patterns directly from the sequence of words. This capability allows it to better understand the structure and flow of language, leading to improved classification performance. Furthermore, although SVM performs relatively well compared to Naïve Bayes, its reliance on sparse representations limits its ability to fully capture contextual meaning, especially in informal and unstructured text commonly found on social media platforms.

Interestingly, the transformer-based BERT model does not achieve the highest performance in this study, despite its advanced contextual representation capabilities. This outcome can be attributed to several factors, including the relatively small dataset size, class imbalance, and limited training epochs. Transformer models typically require large-scale and well-balanced datasets to fully leverage their pre-trained knowledge and contextual embeddings. In contrast, the dataset used in this study is dominated by a single class, which may bias the learning process and reduce the model's ability to generalize across minority classes. Additionally, the confusion matrix analysis reveals that all models, including LSTM and BERT, struggle to correctly classify underrepresented classes, highlighting the significant impact of data imbalance on model performance. These findings suggest that while advanced models offer strong theoretical advantages, their practical effectiveness is highly dependent on data quality, distribution, and training configuration.

Conclusion

This study presents a comparative analysis of machine learning, deep learning, and transformer-based models for multi-class emotion detection on social media data. The experimental results show that the LSTM model achieves the best performance, demonstrating its effectiveness in capturing sequential and contextual information within text. Traditional machine learning models, particularly SVM, remain competitive but are limited by their reliance on feature engineering techniques. Meanwhile, the transformer-based BERT model does not outperform LSTM in this study, highlighting that advanced architectures do not always guarantee superior performance, especially when applied to small and imbalanced datasets. Furthermore, the results reveal that class imbalance significantly affects classification performance, particularly for minority classes. Overall, this study emphasizes the importance of selecting appropriate models based on dataset characteristics and suggests that deep learning approaches such as LSTM may offer a more robust solution for emotion detection in real-world social media scenarios with limited and imbalanced data.

Declarations

Author Contributions

Conceptualization: D.N. and I.K.; Methodology: I.K.; Software: D.N.; Validation: D.N. and I.K.; Formal Analysis: D.N. and I.K.; Investigation: D.N.; Resources: I.K.; Data Curation: I.K.; Writing Original Draft Preparation: D.N. and I.K.; Writing Review and Editing: I.K. and D.N.; Visualization: D.N.; All authors have read and agreed to the published version of the manuscript.

Data Availability Statement

The data presented in this study are available on request from the corresponding author.

Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

Institutional Review Board Statement

Not applicable.

Informed Consent Statement

Not applicable.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] B. Liu, *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies, vol. 5, no. 1, pp. 1–167, May 2012, doi: 10.2200/S00416ED1V01Y201204HLT016.

[2] A. Pak and P. Paroubek, “Twitter as a Corpus for Sentiment Analysis and Opinion Mining,” in *Proc. Int. Conf. Language Resources and Evaluation (LREC)*, Valletta, Malta, May 2010.

[3] T. Joachims, “Text Categorization with Support Vector Machines,” 1998, doi: 10.17877/DE290R-5097.

[4] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. NAACL-HLT*, vol. 2018, no. October, pp. 1-16, 2019, arxiv.org/pdf/1810.04805.

[6] K. Nigam, A. K. McCallum, S. Thrun, et al., “Text classification from labeled and unlabeled documents using EM,” *Machine Learning*, vol. 39, no. May, pp. 103–134, 2000, doi: 10.1023/A:1007692713085.

[7] G. Forman, “An extensive empirical study of feature selection metrics for text classification,” *Journal of Machine Learning Research*, vol. 3, no. March, pp. 1289–1305, Mar. 2003.

[8] O. Vechtomova, “Introduction to Information Retrieval, by C. D. Manning, P. Raghavan, and H. Schütze,” *Computational Linguistics*, vol. 35, no. 2, pp. 307–309, Jun. 2009, doi: 10.1162/coli.2009.35.2.307.

[9] Y. Kim, “Convolutional Neural Networks for Sentence Classification,” in *Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP)*, vol. 2014, no. October, pp. 1746–1751, 2014, doi: 10.3115/v1/D14-1181.

[10] X. Zhang, J. Zhao, and Y. LeCun, “Character-level Convolutional Networks for Text Classification,” in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 2015, no. September, pp. 1-9, 2015, doi: 10.48550/arXiv.1509.01626.

[11] P. Liu, X. Qiu, and X. Huang, “Recurrent Neural Network for Text Classification with Multi-Task Learning,” in *Proc. Int. Joint Conf. Artificial Intelligence (IJCAI)*, vol. 2016, no. May, pp. 1-7, 2016, doi: 10.48550/arXiv.1605.05101.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention Is All You Need,” in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 2017, no. June, pp. 1-15, 2017, doi: 10.48550/arXiv.1706.03762.

[13] T. Wolf, L. Debut, V. Sanh, and J. Chaumond, “Transformers: State-of-the-Art Natural Language Processing,” in *Proc. Conf. Empirical Methods in Natural Language Processing: System Demonstrations (EMNLP Demos)*, vol. 2020, no. October, pp. 38-45, 2020, doi: 10.18653/v1/2020.emnlp-demos.6.

[14] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “RoBERTa: A Robustly Optimized BERT Pretraining Approach,” arXiv preprint arXiv:1907.11692, vol. 2019, no. July, pp. 1-13, 2019, doi: 10.48550/arXiv.1907.11692.

[15] H. He and E. A. Garcia, “Learning from imbalanced data,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, Sept. 2009, doi: 10.1109/TKDE.2008.239.

[16] N. Japkowicz and S. Stephen, “The class imbalance problem: A systematic study,” *Intelligent Data Analysis*, vol. 6, no. 5, pp. 429–449, Nov. 2002, doi: 10.3233/IDA-2002-6504.

[17] D. Tang, F. Wei, N. Yang, and M. Zhou, “Learning sentiment-specific word embedding for Twitter sentiment classification,” in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics (ACL)*, vol. 1: Long Papers, no. June, pp. 1555–1565, 2014, doi: 10.3115/v1/P14-1146.

[18] E. Cambria, B. Schuller, Y. Xia, and C. Havasi, “New avenues in opinion mining and sentiment analysis,” *IEEE Intelligent Systems*, vol. 28, no. 2, pp. 15–21, Mar.–Apr. 2013, doi: 10.1109/MIS.2013.30.

[19] M. Z. Asghar, A. Khan, S. Ahmad, and M. Qasim, “Lexicon-enhanced sentiment analysis framework using rule-based classification scheme,” *PLOS ONE*, vol. 12, no. 2, pp. 1–22, Feb. 2017, doi: 10.1371/journal.pone.0171649.

[20] M. Buda, A. Maki, and M. A. Mazurowski, “A systematic study of the class imbalance problem in convolutional neural networks,” *Neural Networks*, vol. 106, no. October, pp. 249–259, Oct. 2018, doi: 10.1016/j.neunet.2018.07.011.