Brazilian Journal of Biometrics, **43**, 1–13 (2025)
doi:10.28951/bjb.v43i4.786

BRAZILIAN JOURNAL OF
BIOMΣTRICS
ISSN:2764-5290

ARTICLE

Emotion detection in text data: a comparative study of machine learning algorithms

Ikrama Dayyabu Hayatu,[⋆,1] Sachin Singh [1],[⋆,1] Mustapha Mukhtar Muhammad [1],[1] Rohan Mishra,[2] and Manish Mishra[3]

[1]Department of Mathematics, Sharda University, Greater Noida, Uttar Pradesh, India
[2]IILM University, Greater Noida, Knowledge Park-2, Greater Noida, Uttar Pradesh, India
[3]School of Liberal Studies, UPES, Dehradun, Uttarakhand, India
⋆Corresponding author. Email: ikramadayyaburogo@gmail.com; singhat619@gmail.com

(Received: November 18, 2024; Revised: January 13, 2025; Accepted: March 13, 2025; Published: August 22, 2025)

Abstract

Emotion detection plays a vital role in understanding human sentiments and behaviors across various applications, including customer feedback analysis and mental health monitoring. This research assesses the efficiency of different algorithms for machine learning in detecting emotions in text data. A meticulously curated dataset is utilized for the study. The research compares conventional models like Logistic Regression (LR), Random Forest (RF), Support Vector Machines (SVM), and Naive Bayes (NB) with deep learning models like Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), and Bidirectional Encoder Representations from Transformers (BERT). The performance of each algorithm is assessed using accuracy, precision, recall, and F1 score. BERT exhibits superiority over other models, achieving the maximum accuracy of 0.8867 and F1 Score of 0.8871. CNN and SVM also display commendable performance. While the traditional models perform adequately, they are surpassed by deep learning models, with Naive Bayes showing the lowest metrics. This study underscores the significance of selecting models based on specific application requirements, taking into account factors like interpretability and efficiency. Future research endeavors may explore multimodal approaches, model interpretability, bias reduction, and real-time applications, thereby contributing to the advancement of emotion detection in text.

Keywords: Emotion Detection, Textual Data, Machine Learning, Deep Learning, Comparative Study.

1. Introduction

An emotion is a conscious experience marked by intense mental activity and a strong feeling of either pleasure or displeasure. Emotions are vital in information processing, influencing our attention, perception, thoughts, and behavior. They are a fundamental aspect of human existence and significantly impact all areas of our lives, including our health and well-being (Boutet *et al.*, 2021). Emotion detection is important in various fields such as healthcare, political and economic research, driver fatigue detection, and more. As a result, it is an important field of study in human-computer interaction. Emotion detection contributes to understanding human behavior, enhancing the naturalness of human-computer interaction, and facilitating communication between humans and computers in domains such as electronic commerce, e-learning, and gaming (Chowdary *et al.*, 2023). Achieving effective communication with users is essential for technology, software, and machines to reliably perform their tasks of support, service, products, and systems.

1.1 Background of the Study

In recent years, the volume of text data generated daily has increased significantly due to the widespread use of digital communication. This data encompasses social media posts, customer reviews, emails, and instant messages, which can express an extensive variety of emotions (Kushwaha *et al.*, 2021). Understanding these emotions can offer significant perception into public sentiment, customer satisfaction, and individual well-being. Consequently, emotion detection in text has appeared to be a major research area in the field of natural language processing (NLP).

Emotion detection, also known as emotion classification or affective computing, involves identifying and categorizing the emotions expressed in textual data (Wang *et al.*, 2022). Unlike sentiment analysis, which typically classifies text as positive, negative, or neutral, emotion detection aims to categorize definite emotions involving joy, sadness, anger, fear, surprise, and disgust. This level of granularity can facilitate a deeper understanding of the underlying emotional states conveyed by the text (Zad *et al.*, 2021).

1.2 Problem Statement

Despite advancements in machine learning and NLP, accurately detecting emotions in text remains a challenge. Textual data is often ambiguous, context-dependent, and rife with figurative language, making it difficult to capture the true emotional tone. Various machine learning algorithms have been proposed for this task, ranging from traditional classifiers to advanced deep learning models. However, a comprehensive comparison of these algorithms, considering both their performance and computational efficiency, is lacking.

1.3 Objectives and Scope of the Study

This research aims to evaluate various algorithms for machine learning with their comparison in detecting emotions in text data. The specific objectives are as follows:

- Evaluate the accuracy, precision, recall, F1-score, and computational efficiency of diverse machine learning algorithms.
- Identify the strengths and weaknesses of each algorithm in terms of its ability to handle different types of text data and capture emotional nuances.
- Offer insights and recommendations for selecting appropriate algorithms for specific applications of emotion detection.

This study concentrated on a selected group of machine learning algorithms, encompassing traditional methods which include Logistic Regression (LR), Naive Bayes (NB), Random Forest (RF), and Support Vector Machines (SVM), alongside Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN), and Bidirectional Encoder Representations from Transformers (BERT) from advanced deep learning models. The performance of these algorithms was evaluated using a standardized dataset to ensure a fair and consistent comparison.

1.4 Literature Review

The literature on emotion detection in text has undergone significant evolution over the past few decades. It began with basic sentiment analysis and has now advanced to sophisticated models capable of identifying nuanced emotional states. Initially, traditional machine learning algorithms were predominantly used, but they encountered challenges in dealing with the complexities of language (Hung & Alias, 2023). Recent advancements in deep learning have introduced powerful models that significantly enhance the accuracy and depth of emotion detection. Nevertheless, these deep learning models come with their challenges, such as the requirement for computational resources and interpretability (Chowdary *et al.,* 2023).

Several comparative studies to evaluate the performance of various algorithms for machine learning have been conducted in emotion detection. These studies provide a good understanding of the strengths and weaknesses of different approaches. Jadon & Kumar (2023) presented a comparative study on Convolutional Neural Networks (CNNs) with Deep Neural Networks (DNNs) in detecting emotion from text using TF-IDF as the feature extraction technique. The discussion section concluded that both CNNs and DNNs have advantages and disadvantages for emotion recognition using TF-IDF. The choice between the two architectures depends on factors like available resources, interpretability needs, and characteristics of the text data. Future research will involve experiments with deep learning algorithms and word embedding techniques beyond CNN and DNN, as well as further improvements through rigorous preprocessing, hyperparameter tuning, and customized neural network architectures for emotion detection.

In a separate study conducted by Azam *et al.* (2021), a method to automatically detect emotions in healthcare text data about major diseases in Pakistan was developed. The researchers utilized a novel dataset called EmoHD, comprising 4,202 text samples labeled with six emotion classes and eight disease classes. They trained six supervised machine learning models, achieving an accuracy rate of 82% on unseen data. Additionally, the study established a correlation amid negative emotions and psychological health issues. The paper focused primarily on performing emotion classification on text data related to health, specifically addressing prevalent diseases in Pakistan, and psychological health issues linked to negative emotions using an emotional guidance scale. The authors suggested expanding the EmoHD dataset by incorporating more emotion classes or disease types, extending its coverage to diseases beyond Pakistan, and testing different machine learning models, including supervised and unsupervised approaches. They also proposed incorporating embeddings and deep learning models to enhance their work.

Doma & Pirouz (2020) conducted a comparative analysis of techniques for machine learning in classifying emotional statuses throgh EEG (electroencephalogram) data. Their study highlighted that K-Nearest Neighbor (KNN) produced the most favorable results for the "Liking" emotion, prompting the authors to recommend expanding experiments to encompass other emotions. The primary objective was to investigate neurophysiological mechanisms underlying emotional experiences and identify corresponding brain regions. The authors discover that traditional algorithms for machine learning comprise of KNN, SVM, and decision trees achieved satisfactory performance in emotion recognition from EEG data. To enhance accuracy, they recommended integrating techniques for signal processing such as fast Fourier transform and wavelet transform. They also proposed integrating supplementary datasets, like SJTU emotion EEG (SEED) with the Dataset for emotion analysis using physiological signals (DEAP), to potentially enhance performance. Additionally, they suggested exploring complex deep learning and convolutional neural network models, as these approaches have the potential to improve accuracy compared to the manual feature selection approach employed in their study.

Lora *et al.* (2020) presented comparative research on techniques for machine learning with deep learning in detecting positive and negative emotions from Twitter tweets using the CNN model. The paper presents various models for emotion classification using Twitter data. The most successful model is a CNN with pre-trained word embeddings, achieving an accuracy of 84.1%. In general, the deep learning models outperform the machine learning models. The authors suggest two main avenues for future research: first, applying the models developed in this paper to a Bengali social media dataset, and second, applying the models to other text classification problems, such as spam filtering and depression detection.

Mehta *et al.* (2019) presents a comparative study of feature extraction and classification algorithms for accurately recognizing the intensity of facial emotions. The research aims to enable precise emotion intensity recognition in real-world applications. The paper concludes that the mixture of Local Binary Patterns (LBP) with SVM outperforms other techniques in recognizing emotions across all intensity levels. For future investigations, the authors suggest employing neural networks with multiple hidden layers to effectively handle challenges associated with recognizing the intensity of spontaneous facial emotions, such as head tilt and angle.

Salam & Gupta (2018) concentrated on detecting and recognizing emotions from text data on Twitter using techniques for machine learning techniques including K-means, Naive Bayes, and SVM. The paper assessed the effectiveness of these techniques in detecting emotions from twitter data and outlined future research directions. The authors suggested incorporating the association between sentences, implementing semantic-level parsing in addition to syntactic-level processing, and expanding the Bag of Words to include more words describing emotions.

2. Materials and Methods

In this section, we explain the method used to conduct a thorough comparative study on several machine learning algorithms for emotion detection in text data. This includes dataset selection and preprocessing, the evaluation of machine learning algorithms, and the assessment metrics employed to gauge the models' performance.

2.1 Dataset

The data from ISEAR (International Survey on Emotion Antecedents and Reactions) at https://www.kaggle.com/datasets/faisalsanto007/isear-dataset will be utilized for this study, as it is highly suitable for emotion detection tasks. This dataset comprises labeled text samples that express specific emotions, including anger, fear, joy, and sadness. The decision to use the ISEAR dataset was based on its more than two labeling and range of emotional expressions.

2.2 Preprocessing of Data

Data preprocessing is pivotal in preparing the text data for machine learning algorithms. The following steps was executed:

- **Text Cleaning:** Eliminating noise elements such as special characters, numbers, and punctuation marks that do not affect emotion detection.
- **Tokenization:** Text segmentation into individual words or tokens.
- **Removal of Stop Words:** Words elimination that lack substantial meaning, such as "the", "is", and "and".
- **Stemming/Lemmatization:** Conversion of words form into their base or root for the text normalization. For instance, transforming "running" to "run".
- **Vectorization:** Conversion of text data into numerical representation using techniques such as TF-IDF (Term Frequency-Inverse Document Frequency) or word embeddings like Word2Vec, and GloVe, or embeddings can be achieved, from pre-trained models like BERT.

2.3 Vectorization Techniques and Processed Data Used for Each Model

Different vectorization methods to process the text are based on the type of model.

1. **Traditional Machine Learning Models (Logistic Regression, Random Forest, SVM, Naive Bayes)**
   (a) **Vectorization Technique:** TF-IDF (Term Frequency-Inverse Document Frequency).
   (b) **Processed Data:** `TfidfVectorizer(max_features=5000)` converts text data into numerical vectors. `X_train_tfidf` and `X_test_tfidf` consist of the transformed training and testing datasets.
2. **Deep Learning Models (LSTM and CNN)**
   (a) **Vectorization Technique:** Tokenization and Padding.
   (b) **Processed Data:** `Tokenizer(num_words=max_words)` is applied to tokenize words. `texts_to_sequences` converts text into sequences of integers. padding ensures that all sequences in the `X_train_seq` and `X_test_seq` have the same length, and the data padded to uniform size for the models.
3. **BERT Model**
   (a) **Vectorization Technique:** Pretrained BertTokenizer.
   (b) **Processed Data:** `BertTokenizer.from_pretrained('bert-base-uncased')` is used for tokenization. `tokenizer()` is applied to encode the text management with truncation and padding. Used `SentimentDataset` to format the dataset classes into tensors compatible with `PyTorch`, and the data is now structured within the BERT transformer model.

2.4 Machine Learning Algorithms

This study will assess different algorithms for machine learning algorithms, which are categorised into two: traditional machine learning methods and advanced deep learning models.

2.4.1 Traditional Machine Learning Methods

- **Logistic Regression (LR):** This is a linear model used in binary and multiclass classification tasks.
- **Naive Bayes (NB):** A classifier on the ground of probability that relies on Bayes' theorem and assumes strong independence among features.
- **Random Forest (RF):** This is a method for ensemble learning combining multiple decision trees to enhance classification performance.
- **Support Vector Machines (SVM):** This is an effective classifier that find the maximum separating hyperplane that separates different classes.

2.4.2 Deep Learning Models

- **Bidirectional Encoder Representations from Transformers (BERT):** This is a model based on transformer pre-trained on a vast corpus of text, allowing it to capture bidirectional context effectively.
- **Convolutional Neural Network (CNN):** This is a type of deep neural network widely utilized for analyzing visual imagery and has also demonstrated efficacy in text classification tasks.
- **Long Short-Term Memory (LSTM):** This type of recurrent neural network (RNN) is considered to effectively cover and learn dependencies for long term from sequential data.

2.5 Experimental Setup and Execution Flow

The experiment follows a structured pipeline.

2.5.1 Data Loading & Preprocessing

The dataset is loaded using `pd.read_csv('dataset.csv')`. The `LabelEncoder` is used to convert categorical labels into numerical format. The dataset is then split into training and test subsets with an 80%-20% split.

2.5.2 Feature Engineering

- **Traditional Machine Learning Models:** The text data undergoes TF-IDF transformation using `TfidfVectorizer`.
- **LSTM and CNN Models:** These models require tokenization, sequence conversion, and padding using `Tokenizer` and `pad_sequences` functions.
- **BERT Model:** Tokenization is performed using the pretrained `BertTokenizer`, and the dataset is structured as a `SentimentDataset`.

2.5.3 Model Training & Runtime Measurement

- **Traditional Machine Learning Models (Logistic Regression, Random Forest, SVM, Naïve Bayes):** Training is conducted via `.fit(X_train_tfidf, y_train)`. Runtime is recorded using `time.time()` before and after training.
- **LSTM Model:**
  - An `Embedding` layer facilitates word representation.
  - An `LSTM` layer with dropout regularization is used for performance enhancement.
  - The model is compiled using `categorical_crossentropy` as the loss function.
  - Training is conducted for five epochs with a batch size of 64, and runtimes are recorded.
- **CNN Model:**
  - The model starts with an `Embedding` layer for input processing.
  - Various layers such as Convolutional (`Conv1D`), Pooling (`MaxPooling1D`), and Dense layers are implemented.
  - The model is compiled using `categorical_crossentropy` and optimized with the Adam optimizer.
  - Training is performed for five epochs with a batch size of 64, with runtime measurements.
- **BERT Model:**
  - The `BertTokenizer` is used to convert text into numerical representations.
  - Fine-tuning is conducted using Hugging Face's `Trainer` API over three epochs with a batch size of 8.
  - Runtime is recorded throughout the training process.

2.5.4 Model Evaluation

Predictions are generated as follows:

- Traditional ML models: `.predict(X_test_tfidf)`
- LSTM and CNN models: `np.argmax(model.predict(X_test_seq), axis=1)`
- BERT model: `torch.argmax(torch.Tensor(predictions.predictions), axis=1)`

Evaluation Metrics:

- Accuracy
- Precision
- Recall
- F1 Score

2.5.5 Results & Runtime Summary

Performance metrics and runtime details for each model are stored in dictionaries (`results`, `runtimes`). This experiment systematically compares various machine learning techniques—including traditional and deep learning models—to assess their effectiveness in emotion detection from text.

2.6 Model Parameters for Each Algorithm

Table 1 below provides a concise overview of the key hyperparameters used for each model in the experiment.

Table 1. Summary of Key Model Parameters

Model	Key Parameters
Logistic Regression	`max_iter=200`
Random Forest	`n_estimators=100`
Support Vector Machine (SVM)	`kernel='linear'`
Naïve Bayes	Default parameters
LSTM	`Embedding(5000, 100), LSTM(100, dropout=0.2, recurrent_dropout=0.2), epochs=5, batch_size=64`
CNN	`Embedding(5000, 100), Conv1D(128, 5, activation='relu'), MaxPooling1D(2), epochs=5, batch_size=64`
BERT	`bert-base-uncased, num_labels=len(classes), num_train_epochs=3, batch_size=8`

3. Results and Discussion

In this section, we will present and analyze the performance metrics attained by the several machine learning models used for detecting emotions in text data. The models evaluated in this study consist of LR, NB, RF, SVM, BERT, CNN, and LSTM and assessed the models using different metrics including Accuracy, Precision, Recall, and F1 Score.

Table 2. Summary of model performance metrics

	BERT	CNN	LSTM	SVM	LR	RF	NB
Accuracy	0.8867	0.8761	0.8452	0.8705	0.8487	0.8452	0.7607
Precision	0.8883	0.8775	0.8452	0.8751	0.8558	0.8578	0.8037
Recall	0.8867	0.8761	0.8452	0.8705	0.8487	0.8452	0.7607
F1 Score	0.8871	0.8760	0.8445	0.8709	0.8492	0.8457	0.7603

3.1 Interpretation of Results

The analysis on comparison of various models for machine learning on detecting emotions in text data has yielded insightful findings. The performance of each model as shown in **Table 2** varied based on its strengths and weaknesses, shedding light on crucial aspects regarding their applicability and effectiveness in emotion detection tasks.

BERT demonstrated superior accuracy and F1 Score, highlighting its remarkable ability to classify emotion labels effectively. This success is largely due to BERT's advanced bidirectional transformer structure, which allows it to understand the relationships between words in a sentence better than other models. BERT also showed strong precision and recall, meaning it can reliably identify positive instances while also capturing a substantial number of relevant positives. This effective balance between precision and recall makes BERT a trustworthy option for tasks involving emotion detection. The CNN model demonstrates impressive accuracy and F1 Score, following BERT closely. Its ability to detect local patterns in text via convolutional layers plays a crucial role in its strong performance, especially for handling short to medium-length text data. Moreover, CNN's consistent performance across all metrics, including precision and recall, highlights its effectiveness in a variety of text classification tasks.

SVM archived high levels of precision, signifying its capability to accurately pinpoint true positives. This characteristic renders SVM especially beneficial in situations where false positives incur significant expenses. With its strong accuracy and F1 Score, SVM continues to be a top choice for emotion detection, particularly when the data is distinctly separated. Conversely, the LSTM model did not meet expectations, likely due to the dataset's size or the absence of temporal dependencies in the short text samples. Its emphasis on sequential data may limit its effectiveness when handling non-sequential emotional expressions in text.

Both Logistics Regression and Random Forest revealed satisfactory performance, with Random Forest slightly outperforming Logistic Regression. Random Forest's ability to handle non-linear relationships and large feature sets effectively, thanks to ensemble learning, gives it an edge. Through its accuracy and F1 Score, Naive Bayes exhibited the lowest performance metrics. However, in real-world text data, its hypothesis of feature independence is often violated, resulting in less-than-optimal outcomes.

3.2 Comparative Analysis

Figure 1 shows the visual comparison of the models across the evaluated metrics.

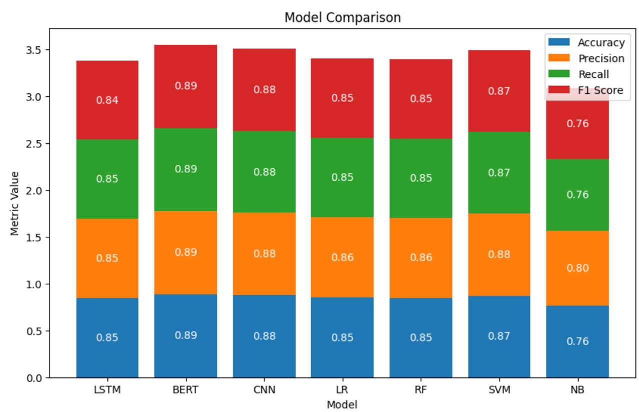

Figure 1. Comparison of model performance metrics.

BERT outperformed all the models compared in this study in all metrics, demonstrating high accuracy, precision, recall, and F1 score. This indicates its highly effective performance in emotion detection for text data. The pre-trained embeddings and fine-tuning capabilities of BERT enable it to capture complex patterns in text data, resulting in superior performance. CNN performed exceptionally well, particularly in terms of accuracy and recall. Its convolutional layers allow it to capture local patterns in text, making it a strong contender for text classification tasks.

SVM demonstrated robust performance with high precision and recall. Its margin maximization principle aids in effective classification, making it a reliable choice for emotion detection. LSTM demonstrated its capability to capture sequential dependencies by achieving a balanced F1 score of 0.8445. While it fell slightly short of CNN and SVM in overall performance, it still showed competitive results, highlighting its potential for emotion detection tasks.

Both Random Forest and Logistic Regression models performed adequately, with Random Forest displaying slightly better metrics than Logistic Regression. Their performance suggests that traditional machine learning models can still be effective for text classification tasks, although they performed less compared to the deep learning models. Naive Bayes performed less among the models. While it is a simple and efficient model, its strong independence assumptions may not hold for complex text data, resulting in lower accuracy and precision.

3.3 Analysis of Model Runtimes

The run time for each model provides insight into its computational efficiency and complexity. The brief analysis of the results is presented in **Table 3** below:

Table 3. Summary of Model Runtime

Model	Runtime (seconds)
LR	0.877373
RF	9.972649
SVM	6.540462
NB	0.005331
LSTM	189.420414
CNN	38.973242
BERT	19081.472364

Naïve Bayes is the fastest but sacrificed accuracy because of its simple probabilistic nature. Logistic Regression is fairly quick, though still performs commendably well. Random Forest and SVM are slower since they resort to ensemble learning and complex optimization.

LSTM and CNNs take much more time since they need training on sequences. BERT has the largest runtime-that's more than 5 hours-owing to the very complexity of transformer-based models.

In this analysis, computation efficiency and model complexity reflect the interplay in emotion detection tasks.

3.4 Visual Plot of the Model Runtime

Figure 2 displays training runtime (in seconds) of different machine learning models utilized in this study. The x-axis comprises the models, specifically Logistic Regression, Random Forest, SVM, Naïve Bayes, LSTM, CNN, and BERT, and the y-axis designates random run times in seconds.

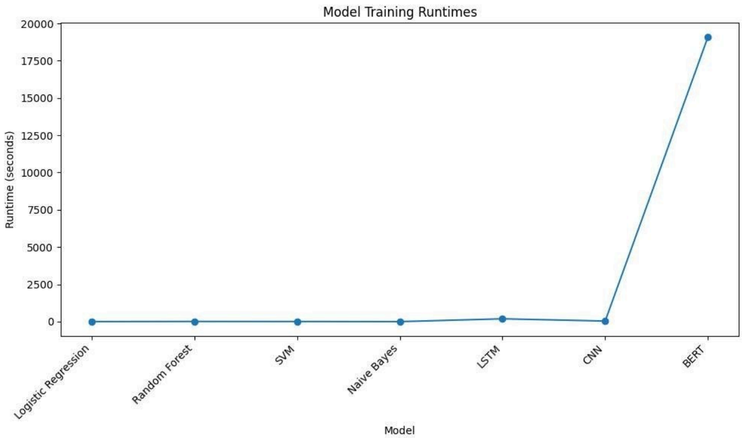

Figure 2. Model's Runtime Plot.

From the line plot, traditional models (Logistic Regression, Random Forest, SVM, and Naïve Bayes) have relatively short training times by contrast to deep learning models (LSTM, CNN). CNN involves a little more training. However, the runtime taken by BERT is exceedingly high, surpassing all other models by a large extent. This signifies that although transformer-based models like BERT give better accuracy, they also require fairly substantial computing power.

3.5 Practical Implications

The findings from this study have implications for using machine learning models for emotion recognition from textual data in the practical aspect.

- **Model Selection for Emotion Detection:** BERT is the recommended model for applications that demand high accuracy and reliability. Its capability to comprehend intricate context and subtle nuances in text makes it well-suited for sophisticated natural language processing tasks. CNN offers a satisfactory balance between performance and computational efficiency. It is appropriate for scenarios where capturing local patterns in text is crucial and computational resources are limited.

- **Resource Allocation:** Organizations must take into account the computational resources and time required for training and deploying models. While BERT delivers exceptional performance, it necessitates significant computational power and longer training times. CNN and SVM deliver competitive performance with the potential for lower resource requirements.
- **Application Context:** In situations where model interpretability is important, simpler models like Logistic Regression and Random Forest may be preferred, even though they have lower accuracy. These models are easier to explain and comprehend, which can be significant in applications such as sentiment analysis for customer feedback.
- **Handling Class Imbalance:** The high precision and recall of BERT and CNN indicate that these models are well-suited for managing imbalanced datasets, where accurately detecting minority class instances is important.

3.6 Model's Strengths and Weaknesses

- The simple and fast model is Naïve Bayes which works well with small datasets and performs well with high-dimensional data. Nevertheless, Naive Bayes is a simple and fast algorithm. It performs well with small datasets and is particularly effective in handling data with higher dimensions.
- The simple and efficient method that works fine with linearly separable data that can easily be interpreted is known as Logistic Regression. However, Logistic Regression assumes a linear decision boundary, which means it may not perform well with complex data patterns.
- Random Forest is capable of handling non-linear data well. It is also robust to overfitting and can handle large feature sets. On the other hand, Random Forest can be computationally intensive. Additionally, it is less interpretable compared to simpler models.
- SVM is effective in high-dimensional spaces and is robust to overfitting when proper regularization is applied. However, it can be expensive computationally, particularly when dealing with large datasets. Additionally, SVM is less effective when confronted with noisy data.
- BERT offers state-of-the-art performance. Its pre-trained embeddings capture complex patterns, and fine-tuning improves accuracy. Nevertheless, BERT is highly computationally intensive and requires significant resources. Training and inference times are also longer.
- CNN is effective at capturing local patterns. It also has relatively faster training compared to LSTMs and is robust to varying input lengths. However, CNN is less effective at capturing long-term dependencies. It can also be less interpretable compared to simpler models.
- LSTM is skilled at capturing long-term dependencies. It is effective for sequential data and is robust to varying sequence lengths. However, we must remember that LSTM can be computationally intensive. It requires large datasets for good performance and longer training times.

4. Conclusions

In this comparative study, we assess the efficiency of different machine learning models in detecting emotions in text data. Among these models, BERT emerges as the most reliable, followed by CNN and SVM. Although traditional models like Logistic Regression and Random Forest demonstrate satisfactory performance, they are surpassed by deep learning models. Naive Bayes, despite being simple and efficient, demonstrates limitations in handling the complexity of emotion detection in text. These findings suggest that advanced models should be preferred for tasks requiring high accuracy and reliability in emotion detection from text data.

4.1 Future Directions

Future research should prioritize several areas, including multimodal emotion detection, transfer learning techniques, interpretability of deep learning models, cross-domain generalization, and real-time applications. It is essential to address biases, ethical considerations, and user-centric design to ensure the responsible deployment and adoption of emotion detection technologies. Additionally, optimizing hyperparameters and experimenting with different model architectures may lead to further improvements in performance.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Conceptualization:HAYATU, I. K., SINGH, S., MUHAMMAD, M. M., MISHRA, R., MISHRA, M. **Data curation**: HAYATU, I. K., SINGH, S., MUHAMMAD, M. M., MISHRA, R., MISHRA, M. **Formal analysis**: HAYATU, I. K., SINGH, S., MUHAMMAD, M. M., MISHRA, R., MISHRA, M. **Funding acquisition**: HAYATU, I. K., SINGH, S., MUHAMMAD, M. M., MISHRA, R., MISHRA, M. **Investigation**:HAYATU, I. K., SINGH, S., MUHAMMAD, M. M., MISHRA, R., MISHRA, M. **Methodology**:HAYATU, I. K., SINGH, S., MUHAMMAD, M. M., MISHRA, R., MISHRA, M. **Project administration**: HAYATU, I. K., SINGH, S., MUHAMMAD, M. M., MISHRA, R., MISHRA, M.**Software**: HAYATU, I. K., SINGH, S., MUHAMMAD, M. M., MISHRA, R., MISHRA, M. **Resources**: HAYATU, I. K., SINGH, S., MUHAMMAD, M. M., MISHRA, R., MISHRA, M. **Supervision**: HAYATU, I. K., SINGH, S., MUHAMMAD, M. M., MISHRA, R., MISHRA, M. **Validation**: HAYATU, I. K., SINGH, S., MUHAMMAD, M. M., MISHRA, R., MISHRA, M. **Visualization**: HAYATU, I. K., SINGH, S., MUHAMMAD, M. M., MISHRA, R., MISHRA, M. **Writing - original draft**: HAYATU, I. K., SINGH, S., MUHAMMAD, M. M., MISHRA, R., MISHRA, M. **Writing - review and editing**: HAYATU, I. K., SINGH, S., MUHAMMAD, M. M., MISHRA, R., MISHRA, M.

References

1. Azam, N., Ahmad, T. & Haq, N. U. Automatic emotion recognition in healthcare data using supervised machine learning. *PeerJ Computer Science* **7,** e751 (2021).
2. Boutet, I., LeBlanc, M., Chamberland, J. A. & Collin, C. A. Emojis influence emotional communication, social attributions, and information processing. *Computers in Human Behavior* **119,** 106722 (2021).
3. Chowdary, M. K., Nguyen, T. N. & Hemanth, D. J. Deep learning-based facial emotion recognition for human–computer interaction applications. *Neural Computing and Applications* **35,** 23311–23328 (2023).
4. Doma, V. & Pirouz, M. A comparative analysis of machine learning methods for emotion recognition using EEG and peripheral physiological signals. *Journal of Big Data* **7,** 18 (2020).
5. Hung, L. P. & Alias, S. Beyond sentiment analysis: A review of recent trends in text based sentiment analysis and emotion detection. *Journal of Advanced Computational Intelligence and Intelligent Informatics* **27,** 84–95 (2023).
6. Jadon, A. K. & Kumar, S. *A Comparative Study of CNNs and DNNs for Emotion Detection from text using TF-IDF* in *2023 International Conference on Advances in Computation, Communication and Information Technology (ICAICCIT)* (2023), 1329–1334.
7. Kushwaha, A. K., Kar, A. K. & Dwivedi, Y. K. Applications of big data in emerging management disciplines: A literature review using text mining. *International Journal of Information Management Data Insights* **1,** 100017 (2021).

8. Lora, S. K., Sakib, N., Antora, S. A. & Jahan, N. A comparative study to detect emotions from tweets analyzing machine learning and deep learning techniques. *International Journal of Applied Information Systems* **12,** 6–12 (2020).
9. Mehta, D., Siddiqui, M. F. H. & Javaid, A. Y. Recognition of emotion intensities using machine learning algorithms: A comparative study. *Sensors* **19,** 1897 (2019).
10. Salam, S. A. & Gupta, R. Emotion detection and recognition from text using machine learning. *Int. J. Comput. Sci. Eng* **6,** 341–345 (2018).
11. Wang, Y. *et al.* A systematic review on affective computing: Emotion models, databases, and recent advances. *Information Fusion* **83,** 19–52 (2022).
12. Zad, S., Heidari, M., James Jr, H & Uzuner, O. *Emotion detection of textual data: An interdisciplinary survey* in *2021 IEEE World AI IoT Congress (AIIoT)* (2021), 0255–0261.