Cross-Domain Sentiment Analysis: An Extensive Study of Machine Learning and Deep Learning Models, Datasets and Preprocessing Techniques for Predictive Performance

Sahil Chordia[1], Dr. Jaya Gupta[2] and Shubham Jain[3]

[1,3] Department of Computer Engineering, A.P Shah Institute of Technology, Thane, India
[2] Department of Computer Science & Engineering (AI & ML), A.P Shah Institute of Technology, Thane, India
chordiasahil24@gmail.com
jdgupta@apsit.edu.in
shubhamjainn1256@gmail.com

Abstract. This extensive and thorough research paper delves deeply into the intricate domain of sentiment analysis within the broader context of natural language processing (NLP). Employing a wide range of advanced machine learning and deep learning models, including Naïve Bayes, Logistic Regression, Random Forest, XGBoost, LSTM, BiLSTM, CNN, and DNN, it meticulously explores sentiment across diverse datasets, encompassing social media content, product reviews, and movie critiques. The research commences with a robust data preprocessing pipeline, addressing aspects such as handling missing values, text case normalization, punctuation removal, tokenization, and target label transformation. It then proceeds to conduct a rigorous evaluation of model performance, employing a diverse set of performance metrics that go beyond accuracy, including precision, recall, and F1-score. The study expands its horizons to uncover sentiment patterns in various domains, from decoding social media reactions to assessing product sentiment for marketing strategies. Furthermore, it critically examines each model's strengths, limitations, and ethical considerations in the realm of sentiment analysis, contributing to a nuanced understanding of the field's practical applications and ethical dimensions. In essence, this comprehensive research illuminate sentiment analysis as a vital tool in data-driven decision-making, underscoring its relevance and ethical implications across diverse industries.

Keywords: Sentiment Analysis, Machine Learning, Deep Learning, Natural Language Processing (NLP), Model Evaluation

1 Introduction

Sentiment analysis, a branch of natural language processing (NLP), is the computational process of deciphering emotional tone, opinions, and sentiments within text data. It involves categorizing text into sentiments such as positive, negative, or neutral. In today's digital era, where a staggering volume of textual data is generated daily, sentiment analysis has become a pivotal tool. Due to the evolution of web 1.0 to web 2.0, it is easiest for users to generate and share their thoughts, views and approaches on the Internet. This leads to an explosive growth of subjective information (opinion data) on the Web. Hence, the objective of gathering, examining, and utilizing this data has grown, leading to the emergence of various concepts, including Sentiment Analysis (SA), which primarily concerns itself with the extraction and categorization of opinions from textual data [1].

The principal aim of Sentiment Analysis is to classify the polarity of textual data, whether it is positive, negative, or neutral. Sentiment Analysis tools empower decision-makers to monitor shifts in public or customer sentiment related to entities, actions, goods, technologies, and services [4]. Twitter sentiment analysis is one of recent and challenging research area Since social media platforms like Twitter contain vast amounts of textual data in the form of tweets, they prove valuable in discerning the sentiments and opinions of individuals regarding particular events [8].

Sentiments can be conveyed in a variety of ways. It can be conveyed by a variety of feelings, judgments, vision or insight, or people's points of view. In several ways, sentiment analysis is carried out. The choice of the sentiment analysis level depends on the available time for the analysis and its relevance to a current task [9]. Sentiment analysis has undergone remarkable evolution, primarily driven by advances in natural language processing and machine learning. It has transitioned from rudimentary rule-based systems to sophisticated models capable of capturing intricate contextual nuances. This transition has been facilitated by the availability of extensive labeled datasets and the rise of pretrained language models. Pretrained models like BERT and GPT have further elevated sentiment analysis by leveraging vast textual corpora and fine-tuning for specific sentiment analysis tasks. These developments have contributed to improved accuracy and scalability. In the realm of sentiment analysis, existing studies often focus on specific domains, limiting their applicability to broader contexts. Unlike these approaches, our paper bridges this gap by employing a novel methodology that transcends domain-specific constraints. We demonstrate the versatility of sentiment analysis models trained on generic Twitter data, showcasing their effectiveness across diverse domains.

In the following sections of this paper, we will provide a detailed overview of our methodology, including data selection and preprocessing techniques, as well as the machine learning and deep learning models employed in our study. Furthermore, we will present the results of our research, showcasing the performance of these models on diverse datasets, with a focus on generic Twitter data, and their real-world applications. Finally, the conclusion will summarize our key findings and their implications for sentiment analysis in various domains and applications.

2 Literature Review

In this section we discuss the related work that have been taken in the field of sentimental analysis using machine learning, deep learning and natural language processing. With the large availability of blogs and social network websites many researchers have delved into sentimental analysis. In a paper on Sentiment Analysis Using Machine Learning Algorithms by Jemai et al [1] the authors tackle the contemporary challenge of sentiment analysis in user-generated content. It benefits from its relevance, dataset accessibility, and algorithm comparisons, claiming to achieve high accuracy. Furthermore, it provides valuable insights for future research directions. However, its primary focus on binary sentiment classification may overlook nuances. The relatively small dataset size limits model complexity, and the paper lacks a comprehensive evaluation. Additionally, practical applications and ethical considerations within sentiment analysis remain unexplored.

A paper by Tusar and Islam [3] investigates sentiment analysis in the US airline industry by employing NLP and machine learning techniques to enhance customer satisfaction assessment. However, it lacks a comprehensive discussion of data preprocessing, evaluation metrics, and NLP techniques, and it doesn't provide a comparison with deep learning methods. Furthermore, the study's generalization and feature engineering aspects are limited, and it does not offer suggestions for future research.

In their research paper, Chandra and Jana [4] present the advantages of enhanced sentiment classification accuracy, enabling organizations to better understand customer opinions and make informed decisions. Furthermore, it harnesses the power of deep learning to uncover intricate data patterns, potentially leading to more insightful insights. Furthermore, it harnesses the power of deep learning to uncover intricate data patterns, potentially leading to more insightful insights. However, the paper lacks in-depth explanations of the algorithms employed, comprehensive performance evaluations across various datasets, and considerations of computational resource requirements, which are crucial for assessing practical applicability. These drawbacks hinder the paper's potential to provide a comprehensive and actionable solution for sentiment analysis challenges in real-world scenarios.

The research paper by K.K Mohbey [19], "A Comparative Analysis of Sentiment Classification Approaches Using User's Opinion" explores sentiment analysis techniques for classifying opinions expressed in tweets and reviews. It evaluates methods like Naïve Bayes, Bernoulli, and Support Vector Machine (SVM) on a dataset of 2000 movie reviews, focusing on accuracy, training time, and prediction time. SVM emerges as the most accurate approach, with potential for broader applications and the consideration of emoji characters in future research.

Similar work by Zhang et al [5] and [9], one is on the survey of Deep learning for sentimental analysis which discusses the rise of deep learning in sentiment analysis due to the abundance of opinionated data on the web. It acknowledges the importance of sentiment analysis in various domains. However, the paper lacks specific details on deep learning techniques, performance evaluations, and potential challenges, limiting its practical applicability. Another one by Shehzadi et al [9] Review on sentimental analysis and emotion detection test tells us that this paper delves into the significance of sentiment analysis and emotion detection within the realm of social media data, underscoring the essential role of swift unstructured data processing. However, it falls short in providing comprehensive methodologies and solutions for these analyses, merely touching on challenges without offering substantial depth or specific insights.

A paper by Vicari and Gaspari [6] in this research paper delves into sentiment analysis of financial news headlines using deep learning, aiming to forecast market sentiment for the DJIA from 2008 to 2020. While the study offers insights, it encounters challenges like limited predictive accuracy and a narrow focus on a single market index. It also neglects a comprehensive analysis of sentiment complexities and the potential risks associated with implementing deep learning in financial trading, raising doubts about its practicality in real-world investment strategies.

In another work, by Kawade et al [8] it focuses on analyzing tweets related to the Uri attack using text mining and R programming. It reveals that 94.3% of people expressed disgust towards the Uri attack, highlighting the negative sentiment associated with the event. The paper emphasizes the significance of sentiment analysis in gauging public opinions on specific events and employs R's text mining packages for the analysis. However, it is constrained by its analysis of only 5000 tweets and suggests future potential in applying big data analysis to larger datasets.

The research paper by Dhabekar et al [10] investigates sentiment analysis using machine learning and deep learning techniques, emphasizing its relevance in understanding public opinions and product/service reviews. However, the paper's drawbacks include a lack of specific details on algorithms and preprocessing methods, limited comprehensive evaluation with essential metrics, dated references, insufficient discussion of dataset selection criteria, and a lack of in-depth exploration of potential limitations, which could impact the study's replicability and robustness.

Combining various methods of deep learning and a comparative study to get the better results. A paper by Dang NC et al [12] examines the application of deep learning in sentiment analysis on social networks like Twitter and Facebook. It reviews recent studies using deep learning models, conducts a comparative analysis, but lacks detailed methodology, specific technique explanations, and overlooks ethical considerations, while also needing more up-to-date references.

The research paper by Ahmed et al [16] introduces a method for detecting cyberbullying in Bangla and Romanized Bangla YouTube comments using NLP and machine learning. The study achieves high accuracy with Multinomial Naive Bayes and Support Vector Machine classifiers, offering potential solutions for early cyberbullying detection. Future work includes scaling up data and expanding the research to diverse video categories.

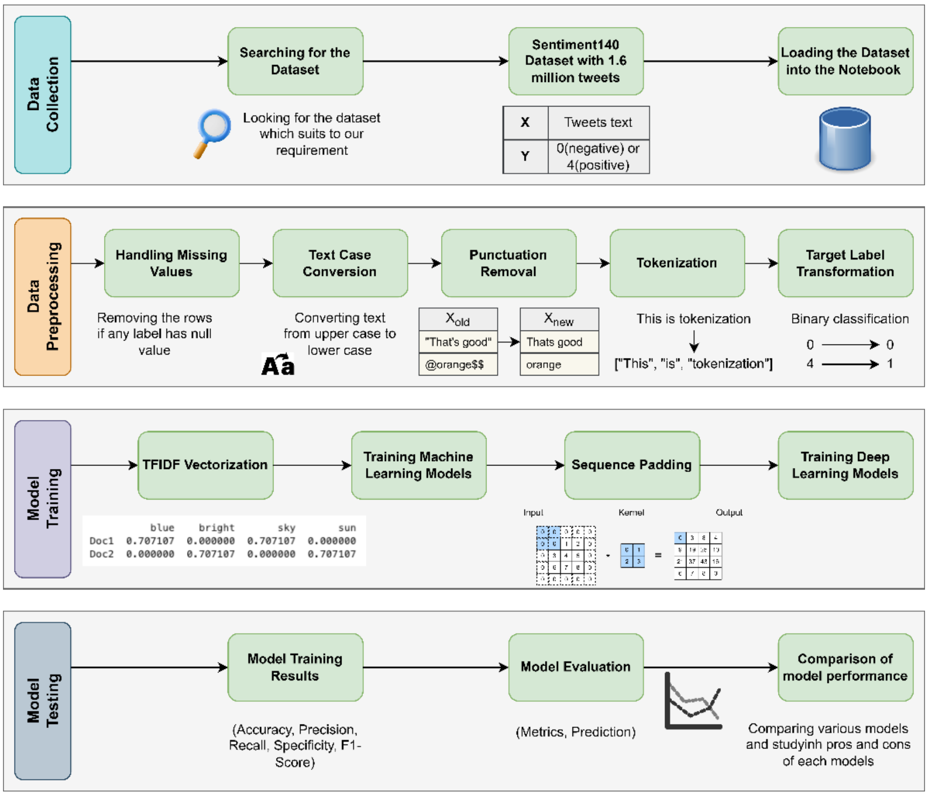
A paper utilizing different methods of deep learning for sentimental analysis by C p.c et al [13], The research paper proposes a sentiment analysis system for Twitter messages using deep learning techniques, particularly LSTM (Long Short-Term Memory) and RNN (Recurrent Neural Network). It classifies tweets into positive/negative emotions and further subcategories, achieving high accuracy rates, with LSTM outperforming RNN. The study incorporates feature selection methods like TF-IDF and Doc2Vec for improved performance. LSTM's robustness in handling long sequences and addressing vanishing gradient problems contributes to its superior performance compared to RNN, making it a promising approach for Twitter sentiment analysis.

A paper by Islam et al [17] This research paper introduces a method to detect sexual harassment in Bangla text on social media using machine learning and deep learning techniques, achieving an accuracy of 89% with CNN-LSTM. It overcomes the limitations of previous studies by focusing on Bangla text. Future work includes dataset expansion and considering Romanized Bangla text.

In another work by K.K Mohbey [18] focuses on sentiment analysis for product ratings using a deep learning approach, specifically Long Short-Term Memory (LSTM) models. The study demonstrates that LSTM outperforms traditional machine learning methods like Naïve Bayes, Support Vector Machine (SVM), and Decision Tree, achieving an accuracy of 93.66%. The research suggests the potential for hybrid approaches and further improvements in sentiment analysis for large datasets.

While the existing literature provides valuable insights into sentiment analysis, it is evident from the surveyed papers that there exists a gap in addressing the challenges of applying sentiment analysis models across diverse domains. Most studies focus on specific industries or domains, limiting their generalizability. Our research strategically bridges this gap by presenting a comprehensive approach that evaluates machine learning and deep learning models on generic Twitter data and subsequently tests their performance on varied domains such as Amazon product reviews and IMDb movie reviews. By addressing the limitations observed in previous works, our paper contributes to the field by offering a more versatile and widely applicable sentiment analysis solution.

3 Methodology


Data Collection
Searching for the Dataset
Sentiment140 Dataset with 1.6 million tweets
Loading the Dataset into the Notebook
Looking for the dataset which suits to our requirement
X
Tweets text
Y
0(negative) or 4(positive)
Data Preprocessing
Handling Missing Values
Text Case Conversion
Punctuation Removal
Tokenization
Target Label Transformation
Removing the rows if any label has null value
Converting text from upper case to lower case
Aa
X_{old}
"That's good"
@orange$$
X_{new}
Thats good
orange
This is tokenization
["This", "is", "tokenization"]
Binary classification
0 → 0
4 → 1
Model Training
TFIDF Vectorization
Training Machine Learning Models
Sequence Padding
Training Deep Learning Models
Input
Kernel
Output
Model Testing
Model Training Results
Model Evaluation
Comparison of model performance
(Accuracy, Precision, Recall, Specificity, F1-Score)
(Metrics, Prediction)
Comparing various models and studyinh pros and cons of each models

```
          blue    bright       sky       sun
Doc1  0.707107  0.000000  0.707107  0.000000
Doc2  0.000000  0.707107  0.000000  0.707107
```

Fig. 1. Block diagram for detailed overview of the methodology

In the diagram presented above (see Fig. 1), we have provided a concise summary of the data selection process and the preprocessing techniques employed. Additionally, the diagram illustrates the utilization of TFidf vectorization for training machine learning models and sequence padding for deep learning models. Finally, the models underwent evaluation and testing on diverse datasets, allowing us to compare the performance of different machine learning and deep learning models.

3.1 Data Creation

The Sentiment140 dataset, obtained from Kaggle [14], plays a pivotal role in our research, facilitating comprehensive sentiment analysis. This extensive dataset encompasses a substantial collection of generic Twitter data, making it ideal for sentiment analysis tasks across various domains and topics. Comprising a vast corpus of 1,600,000 tweets, this dataset offers an invaluable resource for training and evaluating sentiment analysis models. Within the dataset, we primarily leverage two key columns for our research: the tweet text and the target polarity, where polarity values of 0 represent negative sentiment and 4 represent positive sentiment.

3.2 Data Preprocessing

The preprocessing of the sentiment140 dataset was carried out to ensure data quality and consistency for subsequent analysis.. This section outlines the key preprocessing steps applied to the dataset.

Step 1: Handling missing values: The initial dataset was examined for missing values. However, it was observed that there were no instances of missing values present in the dataset. Consequently, no rows required removal due to missing data.

Step 2: Data Cleaning: To standardize the text data and facilitate analysis, the following data cleaning steps were performed:

- **Text Case Conversion:** The text content of the tweets was converted to lowercase, ensuring uniformity and reducing the complexity of subsequent text processing.
- **Punctuation Removal:** Punctuation marks were removed from the text to eliminate potential noise and inconsistencies.

Step3: Tokenization: Tokenization, the process of splitting text into individual tokens (words or phrases), was applied to the cleaned text. Tokenization is a fundamental step for natural language processing tasks, as it prepares the text for further analysis.

Step 4: Target Label Transformation: The target labels in the dataset initially consisted of two categories: 0 for negative and 4 for positive sentiment. To conduct binary sentiment analysis, the dataset was adapted for a binary classification task. Consequently, the target labels were transformed into binary classes: 0 for negative sentiment and 1 for positive sentiment.

3.3 Training Models

Machine Learning Models

In our research for sentiment analysis, the application of machine learning techniques played a pivotal role. To transform the textual data into a format suitable for analysis, we harnessed the power of Term Frequency-Inverse Document Frequency (TF-IDF) vectorization. This process involved converting the raw text into numerical features that could be ingested by machine learning models. The scikit-learn library's TfidfVectorizer facilitated this transformation.

Naïve Bayes Classifier:

One of the cornerstone models employed in our sentiment analysis endeavor was the Naïve Bayes Classifier. Naïve Bayes classifiers are effective and uncomplicated supervised machine learning algorithms [1].This choice was guided by its well-documented effectiveness in text classification tasks. Leveraging the TF-IDF vectors derived from our training dataset, the Multinomial Naïve Bayes Classifier was meticulously trained. Its ability to handle high-dimensional data and inherent simplicity rendered it an excellent starting point for our analysis.

Our Naïve Bayes model exhibited a commendable accuracy rate of approximately 77.19%.

Logistic Regression:

In the realm of machine learning models harnessed for sentiment analysis, Logistic Regression proved to be another indispensable tool. With the TF-IDF vectors as our foundation, we embarked on training a Logistic Regression model to further investigate its predictive competences.

The model was tailored with careful consideration, adjusting parameters such as the maximum number of iterations to ensure robust training (max_iter=1000), and ensuring reproducibility by setting the random seed (random_state=42).

Our Logistic Regression model exhibited a commendable accuracy rate, surpassing 80.25%.

Random Forest:

In our pursuit of crafting an ensemble of diverse machine learning models for sentiment analysis, we introduced the Random Forest model, a robust ensemble learning technique known for its versatility and high predictive performance.

The Random Forest model was constructed using the scikit-learn library with the following key parameters:

- Number of Estimators (n_estimators): 500
- Minimum Samples per Leaf Node (min_samples_leaf): 2
- Out-of-Bag (OOB) Score Calculation (oob_score): Enabled
- Parallel Processing (n_jobs): Utilized all available cores (-1)

During the model training phase, the Random Forest model ingested the preprocessed textual data represented as TF-IDF vectors. The model demonstrated remarkable predictive power, achieving an accuracy of approximately % on the test dataset.

The Random Forest model's architecture thrives on the aggregation of decision trees, each trained on a bootstrapped subset of the data. This ensemble approach results in a robust and highly generalizable model, capable of handling complex relationships within the data.

XGBoost:

In our pursuit of enhancing sentiment analysis accuracy, we explored the application of gradient boosting techniques, specifically using the XGBoost library. XGBoost is a powerful ensemble learning algorithm that has gained popularity for its efficiency and effectiveness in various machine learning tasks.

To harness the capabilities of XGBoost for sentiment analysis, we followed a structured approach:

Data Preparation: We converted our dataset into a DMatrix, the internal data structure used by XGBoost, to efficiently handle large-scale datasets.

Hyperparameter Tuning: We carefully selected hyperparameters to optimize the model's performance. Key hyperparameters include:

- objective: Set to 'binary:logistic' for binary classification.
- max_depth: Limited to 6 to control the depth of individual trees and prevent overfitting.
- learning_rate: Set to 0.1 to control the step size during gradient boosting.
- subsample and colsample_bytree: Configured to 0.8 to introduce randomness and prevent overfitting.
- eval_metric: Chosen as 'logloss' to monitor the model's performance during training.

The XGBoost model was trained with 100 boosting rounds (epochs) using the specified hyperparameters. During training, the model leveraged gradient boosting to iteratively improve its predictive accuracy. After training, the model was used to make predictions on the test dataset. The accuracy of the XGBoost model was evaluated using the accuracy_score metric, resulting in an accuracy of 72.90%.

The XGBoost model, although yielding a slightly lower accuracy compared to other models, contributes to the diversity of our model ensemble and enriches the robustness of our sentiment analysis system.

Deep Learning Models

In our endeavor to elevate the precision of sentiment analysis, we ventured into the domain of deep learning, a formidable subset of machine learning celebrated for its aptitude in discerning intricate data patterns. By harnessing the formidable capabilities of TensorFlow and Keras libraries, we embarked on a transformative odyssey, aiming to harness the latent potential of neural networks in deciphering the complexities inherent to sentiment analysis.

Long Short Term Memory (LSTM)

In our pursuit of deploying advanced deep learning methodologies for sentiment analysis, we delved into the utilization of Long Short-Term Memory (LSTM) neural networks. The Long Short Term Memory network (LSTM) stands as a unique variant of RNNs, possessing the ability to grasp extended-term dependencies [5]. LSTMs are renowned for their proficiency in capturing intricate dependencies within sequential data, making them an apt choice for text-based sentiment analysis.
For the LSTM model architecture, we configured a Sequential model in Keras. The model comprises the following key components:

- **Embedding Layer:** To facilitate the transformation of textual data into numerical vectors, an embedding layer was incorporated. This layer mapped words to high-dimensional vectors, allowing the network to grasp the contextual nuances of the input text. The embedding layer was configured with parameters such as input dimension (vocab_size), output dimension (50), and input sequence length (max_sequence_length).
- **LSTM Layer:** At the heart of the model lies the LSTM layer, which consists of 100 LSTM units. These units enable the model to capture and remember relevant information from the input sequences. LSTM's inherent ability to handle sequence data positions it as an ideal choice for this task.
- **Dense Layer with Sigmoid Activation:** The final layer of the model is a densely connected layer with a sigmoid activation function. This layer yields binary sentiment predictions, with values ranging from 0 (negative sentiment) to 1 (positive sentiment).

In terms of model optimization, we employed the RMSprop optimizer as an alternative to the Adam optimizer, fine-tuning the model's training dynamics.

The model underwent training with a reduced number of epochs (10) and a batch size of 32, adapting to the characteristics of the sentiment analysis task. Subsequently, model evaluation was carried out, and the outcomes revealed an accuracy of 80.57%. The architecture diagram of the LSTM model is presented in Fig 2.

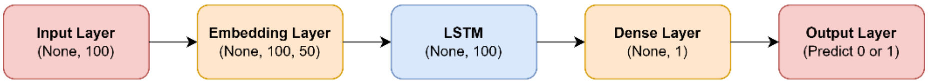


Fig. 2. Block Diagram for Long Short Term Memory (LSTM)

Bidirectional LSTM (BiLSTM):

In our pursuit of achieving higher sentiment analysis accuracy, we explored the utilization of Bidirectional Long Short-Term Memory (BiLSTM) neural networks, a variant of the LSTM architecture that captures contextual information from both the past and the future within the input sequences. This dual-directional memory handling makes

BiLSTMs well-suited for understanding the intricate relationships present in textual data.
The BiLSTM model architecture was structured as follows:

- **Embedding Layer:** Similar to the LSTM model, we employed an embedding layer to convert text data into numerical vectors. This layer mapped words to high-dimensional vectors, enabling the model to capture contextual nuances. Key configuration parameters included input dimension (vocab_size), output dimension (50), and input sequence length (max_sequence_length).
- **Bidirectional LSTM Layer:** The core of the BiLSTM model was the Bidirectional LSTM layer. This layer consisted of 100 LSTM units, and its bidirectional nature allowed it to extract relevant information from both the forward and backward directions of the input sequences. This capability enhanced the model's understanding of sequential data.
- **Dense Layer with Sigmoid Activation:** The final layer of the model was a densely connected layer with a sigmoid activation function. This layer generated binary sentiment predictions, ranging from 0 (indicating negative sentiment) to 1 (indicating positive sentiment).

For model optimization, we adopted the RMSprop optimizer, consistent with the LSTM model. This choice fine-tuned the model's training dynamics.

The BiLSTM model underwent training for a total of 50 epochs, with a batch size of 32, aligning with the sentiment analysis task's requirements. Subsequently, model evaluation was performed, yielding an impressive accuracy of 82.56%. Fig 3 illustrates the architecture diagram of the BiLSTM model.

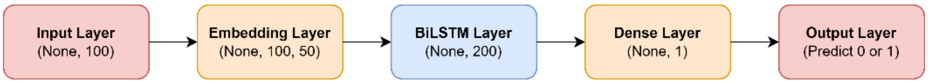


Fig. 3. Block Diagram for Bidirectional LSTM (BiLSTM)

Convolutional Neural Network (CNN):

In our quest to explore various deep learning architectures for sentiment analysis, we introduced a Convolutional Neural Network (CNN) model into our research arsenal. The CNN architecture, renowned for its ability to capture intricate hierarchical patterns within data, played a pivotal role in our study. In a Convolutional Neural Network (CNN), various convolutional layers, coupled with a pooling layer and an output layer, are employed [9].
The architecture of the CNN model can be outlined as follows:

- **Embedding Layer:** The initial layer of our CNN model, known as the Embedding layer, served the critical purpose of translating textual data into numerical vectors. This layer facilitated the encoding of words into high-dimensional representations, enabling the network to discern the contextual nuances within the input text. The Embedding layer featured configurations such as input dimension (vocab_size), output dimension (100), and input sequence length (max_sequence_length).

- **Convolutional and Pooling Layers:** Following the Embedding layer, our model incorporated a Conv1D layer comprising 128 filters and a kernel size of 5. This layer harnessed the power of convolution operations to detect essential features within the input data. Subsequently, a GlobalMaxPooling1D layer was employed to extract the most salient information from the convolutional outputs, enhancing the model's capacity to capture relevant patterns.
- **Densely Connected Layers:** To facilitate complex feature learning, our model featured two densely connected layers equipped with Rectified Linear Unit (ReLU) activation functions. These layers played a crucial role in refining the extracted features, ultimately contributing to the model's ability to discriminate between different sentiment classes.
- **Binary Classification Layer:** The final layer of our CNN model utilized a sigmoid activation function. This configuration resulted in binary sentiment predictions, with values ranging from 0 (indicative of negative sentiment) to 1 (representing positive sentiment).

The CNN model employed the RMSprop optimizer, mirroring the optimization strategy used for LSTM and BiLSTM. This choice was made to ensure consistency across model training dynamics.

The CNN model underwent rigorous training, spanning 10 epochs, with a batch size of 32. Subsequent evaluation unveiled the model's proficiency, yielding an accuracy of 79.96%. The following diagram (see Fig.4) shows the layers used in CNN.

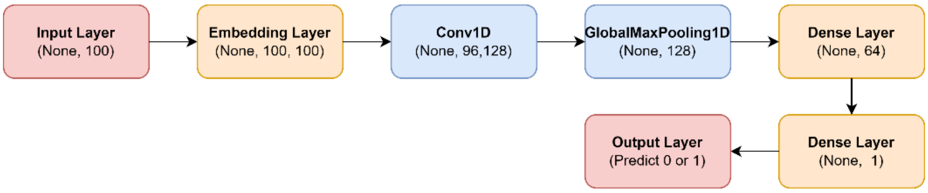


Fig. 2. Block Diagram for Convolutional Neural Network (CNN)

Dense Neural Network (DNN):

As part of our exploration into diverse deep learning architectures for sentiment analysis, we incorporated a Deep Neural Network (DNN) model into our research portfolio. Deep neural networks utilize advanced mathematical models to manipulate data through diverse methodologies[12].The DNN architecture, marked by its stacked layers of neurons, demonstrated its capability in capturing intricate patterns within textual data.

The architecture of the DNN model can be outlined as follows:

- **Embedding Layer:** Similar to the LSTM and BiLSTM models, we initiated our DNN model with an embedding layer. This layer was responsible for transforming the textual data into numerical representations. It mapped words to high-dimensional vectors, allowing the model to discern the intricate semantics of the input text. The configuration parameters included input dimension (vocab_size), output dimension (50), and input sequence length (max_sequence_length).
- **Flatten Layer:** Following the embedding layer, we introduced a flatten layer. This layer reshaped the multidimensional output from the embedding layer into a one-dimensional vector. This flattened representation served as the input for the subsequent dense layers.
- **Dense Layers:** The DNN model comprised multiple densely connected layers. Specifically, it included two dense layers with 128 and 64 units, respectively, and ReLU (Rectified Linear Unit) activation functions. These layers facilitated feature extraction and abstraction, enabling the model to uncover latent patterns within the data.
- **Output Layer with Sigmoid Activation:** The final layer of the DNN model was a densely connected layer with a sigmoid activation function. This layer produced binary sentiment predictions, assigning values between 0 (indicating negative sentiment) and 1 (indicating positive sentiment).

The DNN model employed the RMSprop optimizer, mirroring the optimization strategy used for the other deep learning models. This choice was made to ensure consistency across model training dynamics. During training, the DNN model underwent 10 epochs, with a batch size of 32, in alignment with the sentiment analysis task's requirements. Subsequent model evaluation reported an accuracy of 79.90%. The detailed block diagram has been shown in Fig.5 which demonstrates the layers of the DNN.

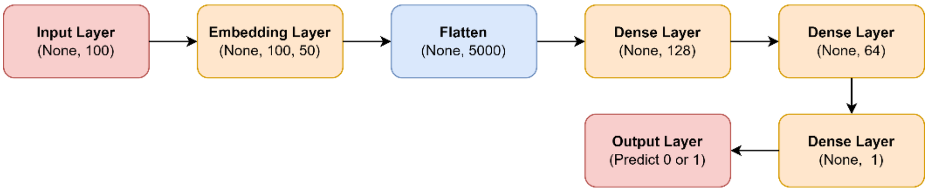


Fig. 5. Block Diagram for Dense Neural Network (DNN)

4 Results And Discussion

Following the rigorous training phase on Dataset 1, the Sentiment140 dataset, our research extended to encompass two additional datasets, Dataset 2 and Dataset 3, acquired from Kaggle [15]. These datasets, comprised of Amazon product reviews and IMDB movie reviews, respectively, were annotated with sentiment labels, with '0' representing negative sentiments and '1' indicating positive sentiments. Our comprehensive evaluation of model performance across these datasets involved the utilization of a uniform preprocessing pipeline similar to that applied to Dataset 1. Subsequently, we conducted a detailed analysis of model efficacy, yielding a range of performance metrics for each model as outlined below:

Table 1. Comparative Analysis of Machine Learning Models (D1, D2, and D3 refers to Dataset1, Datset2 and Datset3 respectively)

ML Models	**Naïve Bayes**			**Logistic Regression**			**Random Forest**			**XG Boost**		
	D1	D2	D3	D1	D2	D3	D1	D2	D3	D1	D2	D3
Accuracy	77.19	71.93	***76.46***	***80.25***	***77.97***	75.75	78.93	73.87	73.74	72.90	63.25	62.58
Precision	80.75	81.36	74.29	79.82	74.29	69.93	81.42	74.60	69.14	70.36	58.80	58.05
Recall	71.59	56.70	80.58	81.13	85.47	89.86	75.15	72.05	85.04	79.51	89.90	89.67
Specificity	82.83	87.07	72.41	79.36	70.51	61.76	82.74	75.61	62.63	66.29	37.20	35.71
F1-Score	75.90	66.33	77.36	80.47	79.40	78.60	78.16	73.61	76.19	74.66	71.10	70.15

Table 2. Comparative Analysis of Deep Learning Models(D1, D2, and D3 refers to Dataset1, Datset2 and Datset3 respectively)

DL Models	**LSTM**			**CNN**			**BiLSTM**			**DNN**		
	D1	D2	D3	D1	D2	D3	D1	D2	D3	D1	D2	D3
Accuracy	80.57	80.74	79.28	79.96	***82.17***	78.97	***82.56***	80.43	***79.48***	79.90	77.77	75.55
Precision	80.46	80.33	75.68	84.20	88.66	80.00	82.81	78.99	75.68	80.83	76.53	70.89
Recall	80.91	81.34	85.93	73.92	73.76	76.90	81.71	82.93	86.44	78.55	79.88	86.44
Specificity	80.23	80.16	72.55	86.04	90.57	82.06	82.92	77.91	72.55	81.26	75.61	64.77
F1-Score	80.68	80.83	80.41	78.73	80.50	78.42	82.26	80.91	80.72	79.68	78.16	77.80

4.1 Performance on different datasets:

In order to comprehensively assess the capabilities and robustness of the machine learning and deep learning models under consideration, we conducted a detailed evaluation of their performance across three diverse datasets (see Table 1 and Table 2). This approach allows us to gain a deeper understanding of how these models perform in various real-world scenarios. Each dataset represents a distinct domain, and the models' performances provide valuable insights into their adaptability and efficacy. In the following sections, we delve into the performance of these models on individual datasets, highlighting their strengths and applicability in different contexts.

Dataset 1 (D1): Sentiment140 Dataset

Highest Accuracy: The BiLSTM model achieved the highest accuracy of 82.56%, closely followed by the LSTM model with an accuracy of 80.57%. These models excel in classifying sentiments in this dataset.

Precision and Recall Balance: The LSTM model achieved a well-balanced precision and recall, making it suitable for applications where both positive and negative sentiments need to be accurately identified.

Robust Performance: The Random Forest model also performed well with an accuracy of 78.93% and a good balance between precision and recall, making it a robust choice for this dataset.

Dataset 2 (D2): Amazon Product Reviews Dataset

Highest Accuracy: The CNN model outperformed other models with an accuracy of 82.17%, indicating its suitability for classifying sentiments in Amazon product reviews.

High Precision: The Logistic Regression model achieved the highest precision of 88.66%, making it suitable for tasks where minimizing false positives is essential.

Varied Performance: While the BiLSTM model maintained good precision and recall, the XG Boost model demonstrated a high recall rate. Depending on the specific application, these models can be considered.

Dataset 3 (D3): IMDB Movie Review Dataset

Highest Accuracy: The BiLSTM model showed the highest accuracy of 79.48%, indicating its effectiveness in classifying sentiments in IMDB movie reviews.

High Recall: The BiLSTM model also demonstrated high recall, suggesting its ability to effectively identify both positive and negative sentiments.

Balanced Performance: The LSTM model achieved a good balance between precision and recall, making it a reliable choice for this dataset

4.2 Data Influence:

The variance in model performance across datasets underscores the influence of data characteristics on model outcomes. For instance, Dataset 2, consisting of Amazon product reviews, presents a unique challenge with mixed sentiments and nuanced language. Models that perform well on this dataset may have a better understanding of context and sentiment subtleties.

4.3 Deep Learning vs. Traditional ML:

Comparing deep learning models like BiLSTM and CNN to traditional machine learning models like Logistic Regression and Random Forest, we observe that deep learning models generally perform better in capturing intricate patterns in text data. However, they may require more extensive computational resources and data for training.

4.4 Model Selection for Real-world Applications:

The choice of model should align with the specific requirements of the application.

- **BiLSTM:** Suitable for applications requiring a balanced approach between precision and recall, such as sentiment analysis for product reviews, where both accurate identification of positive sentiment and capturing nuanced negative sentiment are important.
- **Logistic Regression:** Effective for sentiment analysis in scenarios where a straightforward and interpretable model is preferred. It can be used in e-commerce platforms to classify customer reviews into positive or negative sentiment categories.
- **LSTM:** Ideal for applications demanding high recall, such as social media sentiment analysis, where identifying negative sentiment or potential issues is critical. It can be used to monitor and respond to customer sentiment on social media platforms.
- **CNN:** Useful for sentiment analysis in multimedia content, such as analyzing sentiment in images or videos with textual descriptions. It excels at identifying visual and textual cues for sentiment classification.
- **DNN:** Versatile and well-suited for applications requiring a balanced performance between precision and recall. It can be employed in sentiment analysis for general-purpose text data, where maintaining a good trade-off between precision and recall is essential.
- **Naïve Bayes:** Valuable in resource-constrained environments or for simple sentiment analysis tasks. It can be utilized in applications like email sentiment classification, where speed and simplicity are more critical than nuanced sentiment analysis.
- **XG Boost:** May find applications in situations where ensemble models are favored for enhanced predictive performance. It can be used in sentiment analysis tasks that require high accuracy, like real-time stock market sentiment analysis.
- **Random Forest:** Suitable for applications where ensemble learning is preferred and robustness to noisy data is essential. It can be applied in sentiment analysis for user generated content, such as comments on a news website.

4.5 Practical Implication

In practical deployment, the overall goals and constraints of the application should guide model selection. Real-world factors like data availability, computational resources, and scalability should be considered.

5 Conclusion

In this research paper, we conducted an in-depth exploration of sentiment analysis models, both from the domain of machine learning and deep learning, and their performance on diverse datasets. Our study encompassed a comprehensive analysis of model performance, including accuracy, precision, recall, specificity, and F1-score, across three distinct datasets: Sentiment140, Amazon product reviews, and IMDB movie reviews.

Our findings have shed light on several important aspects of sentiment analysis and model selection. We observed that the choice of model should align with the specific requirements and constraints of the application, considering factors such as precision, recall, interpretability, and resource constraints. Deep learning models, particularly BiLSTM and CNN, demonstrated superior performance in capturing intricate patterns within textual data, making them valuable choices for certain applications.

The performance variation across datasets highlights the influence of data characteristics on model outcomes. Dataset-specific nuances, such as mixed sentiments, context, and language subtleties, played a crucial role in model performance. It is essential for practitioners to carefully consider the nature of their data and the specific goals of their sentiment analysis task when selecting an appropriate model.

The significance of our research extends beyond the specific models and datasets studied. Our findings have practical implications for a wide range of real-world applications, from social media sentiment analysis to e-commerce customer reviews and multimedia content analysis. The selection of an appropriate model can significantly impact the effectiveness of sentiment analysis systems, and our research provides guidance for such decisions.

Further research avenues are evident, including hyperparameter tuning, ensemble methods, and domain-specific pre-trained embeddings. These directions can lead to even more robust and accurate sentiment analysis models, particularly when applied to diverse and challenging datasets.

In conclusion, this research paper contributes to the field of natural language processing and sentiment analysis by offering a comprehensive evaluation of models on diverse datasets and providing practical insights for model selection. Our study equips researchers and practitioners with valuable knowledge to make informed decisions when implementing sentiment analysis systems in various domains and applications.

6 Author Contributions

Sahil Chordia conceived the idea for this research paper and played a pivotal role in its execution. He led the model training efforts for sentiment analysis using machine learning and deep learning techniques. His contribution extends to the comprehensive documentation of the research and setting the strategic goals. His vision and dedication significantly shaped the direction of this work.

Dr. Jaya Gupta, provided invaluable guidance and direction to the research project. Her expertise and mentorship were instrumental in steering the research towards successful outcomes. Dr. Gupta's consistent support and insightful contributions enriched the quality of the research.

Shubham Jain, the third author, made significant contributions to the research by diligently undertaking tasks as assigned. His active involvement in model training and documentation substantially expedited the research process. Shubham's commitment and collaborative spirit were vital in achieving the research objectives.

References

1. F. Jemai, M. Hayouni and S. Baccar, "Sentiment Analysis Using Machine Learning Algorithms," 2021 International Wireless Communications and Mobile Computing (IWCMC), Harbin City, China, 2021, pp. 775-779, doi: 10.1109/IWCMC51323.2021.9498965.
2. Kastrati Z, Dalipi F, Imran AS, Pireva Nuci K, Wani MA. Sentiment Analysis of Students' Feedback with NLP and Deep Learning: A Systematic Mapping Study. *Applied Sciences*. 2021; 11(9):3986. https://doi.org/10.3390/app11093986
3. M. T. H. K. Tusar and M. T. Islam, "A Comparative Study of Sentiment Analysis Using NLP and Different Machine Learning Techniques on US Airline Twitter Data," 2021 International Conference on Electronics, Communications and Information Technology (ICECIT), Khulna, Bangladesh, 2021, pp. 1-4, doi: 10.1109/ICECIT54077.2021.9641336.
4. Y. Chandra and A. Jana, "Sentiment Analysis using Machine Learning and Deep Learning," 2020 7th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2020, pp. 1-4, doi: 10.23919/INDIACom49435.2020.9083703.
5. Zhang, L., Wang, S., & Liu, B. 2018. Deep Learning for Sentiment Analysis: A Survey. arXiv, arXiv:1801.07883.https://doi.org/10.48550/arXiv.1801.07883
6. Vicari M, Gaspari M. Analysis of news sentiments using natural language processing and deep learning. AI Soc. 2021;36(3):931-937. doi: 10.1007/s00146-020-01111-x. Epub 2020 Nov 30. PMID: 33281303; PMCID: PMC7701378.
7. Nandwani, P., Verma, R. A review on sentiment analysis and emotion detection from text. *Soc. Netw. Anal. Min.* **11**, 81 (2021). https://doi.org/10.1007/s13278-021-00776-6
8. Kawade, Dipak & Oza, Kavita. (2017). Sentiment Analysis: Machine Learning Approach. International Journal of Engineering and Technology. 9. 2183-2186. 10.21817/ijet/2017/v9i3/1709030151.
9. Shehzadi, K., & Raza, U. A. (2021). Sentiment Analysis by Using Deep Learning and Machine Learning Techniques: A Review. International Journal of Advanced Trends in Computer Science and Engineering, 10(2), 754-761.
10. Dhabekar, S., & Patil, M. D. (2021). Implementation of Deep Learning Based Sentiment Classification and Product Aspect Analysis. ITM Web Conf., 40, 03032. https://doi.org/10.1051/itmconf/20214003032
11. Balci, S., Demirci, G.M., Demirhan, H., Sarp, S. (2022). Sentiment Analysis Using State of the Art Machine Learning Techniques. In: Biele, C., Kacprzyk, J., Kopeć, W., Owsiński, J.W., Romanowski, A., Sikorski, M. (eds) Digital Interaction and Machine Intelligence. MIDI 2021.

Lecture Notes in Networks and Systems, vol 440. Springer, Cham. https://doi.org/10.1007/978-3-031-11432-8_3

12. Dang NC, Moreno-García MN, De la Prieta F. Sentiment Analysis Based on Deep Learning: A Comparative Study. *Electronics*. 2020; 9(3):483. https://doi.org/10.3390/electronics9030483
13. C p.c, S., Shereen, R., Jacob, S., & Vinod, P. (2021). Sentiment Analysis Using Deep Learning. In 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV) (pp. 1-5). DOI: 10.1109/ICICV50876.2021.9388382
14. https://www.kaggle.com/datasets/kazanova/sentiment140
15. https://www.kaggle.com/datasets/marklvl/sentiment-labelled-sentences-data-set
16. Ahmed, Md.Tofael & Rahman, Maqsudur & Nur, Shafayet & Islam, A. & Das, Dipankar. (2022). Natural language processing and machine learning based cyberbullying detection for Bangla and Romanized Bangla texts. TELKOMNIKA (Telecommunication Computing Electronics and Control). 20. 89-97. 10.12928/telkomnika.v20i1.18630.
17. Islam, Mujahidul & Rahman, Maqsudur & Ahmed, Md.Tofael & Islam, Abu & Das, Dipankar & Hoque, Moshiul. (2023). Sexual Harassment Detection using Machine Learning and Deep Learning Techniques for Bangla Text. 1-6. 10.1109/ECCE57851.2023.10101522.
18. K. K. Mohbey, "Sentiment analysis for product rating using a deep learning approach," 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), Coimbatore, India, 2021, pp. 121-126, doi: 10.1109/ICAIS50930.2021.9395802.
19. Mohbey, Krishna Kumar, A Comparative Analysis of Sentiment Classification Approaches Using User's Opinion (April 20, 2018). Proceedings of 3rd International Conference on Internet of Things and Connected Technologies (ICIoTCT), 2018 held at Malaviya National Institute of Technology, Jaipur (India) on March 26-27, 2018, Available at SSRN: https://ssrn.com/abstract=3166074 or http://dx.doi.org/10.2139/ssrn.3166074
20. Source code can be found on https://github.com/protocorn/sentiment-analysis-using-machine-learning-and-deep-learning-models