International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2022): 7.942

Intelligent Sentiment Prediction in Social Networks leveraging Big Data Analytics with Deep Learning

Maria Anurag Reddy Basani

Texas A & M University, Corpus Christi, USA

Abstract: ***Cloud computing has recently made it easier to distribute varied, unstructured digital data within social networks of differing opinions. Processing large volumes of text data requires precise computational methods, which increases the system's workload. Integrating big data with Natural Language Processing (NLP) has enhanced this. Frameworks like MapReduce enable parallel computation for large tasks. This study proposes a smart sentiment model based on Deep Learning (DL) and batch and streaming big data analytics. The research aim is to utilize the power of distributed platforms for real-time data processing. These platforms assist with data cleaning, size reduction, decreasing access times, and reducing storage needs. This preprocessing step makes the streaming more suitable for data-intensive models. This research focuses on handling large-scale, short-text data using batch and streaming frameworks combined with DL techniques in NLP. We present a method to analyze short texts, determine their semantic meaning, and classify them into positive or negative sentiments. The process involves data reduction and refinement using selected features and big data tools, followed by embedding words with global vectors to feed into convolutional and RNNs. The experimental results confirm the effectiveness of our approach, demonstrating its superiority over existing methods. Our model achieved a remarkable accuracy of 97.31%.***

Keywords: Sentiment Analysis, Multilingual, Big Data, Deep Learning, BERT, Classification Performance, Natural Language Processing, CNN, LSTM, Cross-Language

1. Introduction

1.1 Background

Recently, there has been rapid growth of digital technology, especially in areas like cloud computing (1) and the Internet of Things (IoT) (2). This has led to a significant surge in digital data generation from diverse platforms such as social networks (3). Handling such massive volumes of data with high accuracy requires developing advanced techniques. Among the most prevalent forms of digital interaction is textual communication, which allows users to express opinions. Despite advances in DL, often employed to classify sentiment in the text as either positive or negative. However, existing models encounter several challenges due to massive data (4). These difficulties arise because sentiment classification relies on understanding word semantics, frequency of occurrence, and contextual meaning.

In the context of large-scale decision-making (LSDM) (5), transforming scientific data into actionable insights can play a crucial role in assisting businesses and organizations in making well-informed choices about their products and services. This data-driven approach leads to improved productivity. However, LSDM is inherently complex as it requires a deep understanding of group dynamics, including consensus or conflicts, to promote effective decision-making. Various DL models have been developed to classify the sentiment polarity of both structured and unstructured text, leveraging natural language processing (NLP) techniques (6). However, many of these models need to be more accurate in accurately identifying sentiment and efficiently processing batch and streaming text data.

To address these shortcomings, we have designed a comprehensive strategy built upon the frameworks outlined in (7). Our approach integrates batch processing with real-time streaming data frameworks (8; 9), ensuring scalability and adaptability for large-scale text processing. By applying a range of techniques, including DL architecture and advanced feature selection, our approach improves sentiment prediction across a variety of fields such as sports analysis (10), healthcare data mining, agricultural data insights (11), smart city management (12), and marketing strategies (13).

1.2 Contributions

After thoroughly reviewing recent work in big data analysis, we identified limitations in many existing approaches. Most methods exhibit poor accuracy due to the low quality of training data used for classification. This work introduces a model designed specifically for short texts. The following key innovations distinguish our method:

- We emphasize data preprocessing, including cleaning, size reduction, access time optimization, and storage minimization, before applying DL.
- Big data frameworks like Hadoop and Spark, with modified MapReduce architecture, are used to implement efficient data preprocessing.
- A strategy for analyzing and classifying streaming data is developed using multi-channel CNN.
- Bidirectional LSTM networks are integrated to enhance short-text classification within large-scale data environments.
- The method demonstrates improved evaluation metrics, including precision, recall, and F1-score, outperforming traditional approaches.

Our approach's novelty lies in its multi-step preprocessing pipeline, which includes advanced cleaning algorithms, innovative size reduction techniques, optimized access time mechanisms, and storage volume reduction methods. This pipeline is designed to handle large-scale streaming data with precision and efficiency, providing high-quality inputs to DL models.

1.3 Paper Organization

This paper is structured as follows: Section 2 describes the architecture of big data analysis frameworks and DL techniques. Section 3 reviews previous research on big data analysis using machine learning. Section 4 introduces our advanced system for classifying large-scale data aimed at supporting decision-making processes for companies. The system consists of a preprocessing stage using big data analysis frameworks, followed by a classification stage. Section 5 presents an empirical study to validate our model's performance against other existing approaches. Section 6 discusses the results of our experiments. Finally, Section 7 concludes the paper and outlines directions for future research.

2. Literature Review

The surge in large-scale textual data has led to the need for sophisticated methods to process and analyze this data (14). Such data, often from social networks, e-commerce, and other applications, is critical for improving decision-making in various fields. To address the challenges of processing this vast data, several approaches in big text data analysis have emerged (15). These approaches generally classify text data into positive or negative sentiment poles, allowing researchers to draw valuable insights from user opinions.

One of the key challenges discussed in the literature is identifying consumer sentiment from short-form text data (16). Short texts, commonly found on social media platforms, pose unique challenges due to their brevity and the nuanced context they often carry. Studies focusing on short texts have developed methods using NLP to extract opinions on brands and products. One such study proposed a fake review detection framework that achieved an accuracy of 85.5% (13). However, it faced limitations due to the complexity of the data and its subjectivity.

Researchers have integrated DL models with big data frameworks to tackle the challenges of large-scale text data and improve processing efficiency and accuracy. For instance, the PABIDDL method, which employs DL in combination with Hadoop's MapReduce and GloVe embeddings, was able to classify text into benefit and disadvantage categories with an accuracy of 93% (17). This demonstrates the power of combining machine learning with big data platforms to manage and analyze massive datasets.

Another promising approach is the DeepEmotionNet model, which uses advanced neural architectures such as contextual encoders and embeddings to analyze sentiment. This model improved F-scores by up to 29.8% compared to earlier techniques, illustrating its capability to capture sentiment nuances, particularly in corporate communication and financial contexts (18). Similarly, credibility assessment models, such as CreCDA, which analyzes social media content to determine the trustworthiness of conversations, achieved a credible F1-score of 79% by integrating user and publication features (19).

Beyond sentiment analysis and credibility detection, text classification methods have been applied to various domains. For instance, in product development, a model leveraging Hadoop and CNNs was used to analyze consumer feedback to enhance electric vehicle design, achieving an accuracy of 85% (20). In another example, recurrent neural networks (RNNs), such as LSTM and BiLSTM, were combined with fastText to improve classification accuracy in handling large-scale social data. However, this approach reached an accuracy of only 79% (21).

Word embedding models have also been central to improving text classification. For instance, CBOW (Continuous Bag of Words) has been used in combination with CNNs to analyze commentary texts, achieving accuracy rates between 87.2% and 90.5% across various datasets (22). Furthermore, distinguishing between human generated and bot-generated content on platforms like Twitter has become a focus, with models integrating semantic and stylistic features to classify such content. One such study reported bot detection accuracy of 93.47% and gender detection accuracy of 92.44% (23).

Despite the significant advancements in big text data analysis, the reviewed models still present limitations (24). Many fail to address the need to simultaneously handle multiple datasets or reduce dataset size while maintaining classification accuracy (25). Moreover, the rapid evolution of cloud computing and IoT further exacerbates the need for efficient methods to process ever-growing datasets.

The traditional big data frameworks have played a pivotal role in managing large-scale datasets (26). Their effectiveness could be better by suboptimal access times and lower classification accuracy. Nevertheless, DL methods have exhibited strong performance and efficiency. Integrating these methods with big data frameworks provides the necessary scalability and performance improvements for handling large-scale text data, though challenges in optimizing these frameworks remain.

While significant progress has been made in applying DL models to large-scale text classification, there is a continued need for more advanced frameworks that improve both accuracy and processing efficiency. The integration of DL techniques with scalable big data solutions holds great promise for overcoming the challenges posed by the rapidly expanding datasets generated by modern technological infrastructures.

3. Materials and Methods

We propose a high-level architecture focusing on processing and classifying large-scale textual data using batch and streaming methods. Our approach integrates advanced versions of the Hadoop and Spark Big Data Analytics frameworks in the preprocessing stage and leverages DL models in the classification phase. This allows businesses to utilize user reviews on social media platforms to make better-informed decisions regarding their products and services.

Preprocessing: Big Data Analytics

The first stage of our model involves preprocessing the data using an enhanced MapReduce framework. The improvements focus on feature selection, cleaning, and dimensionality reduction. These modifications to the

MapReduce architecture significantly improve the accuracy and efficiency of the classification tasks performed in later stages. This algorithm modifies the standard MapReduce architecture to include feature selection and optimized parallel processing. The goal is to preprocess the data efficiently and remove irrelevant information before passing it on to the classification phase.

Algorithm 1 Enhanced MapReduce for Preprocessing

Input: $HADOOP_{HDFS}$ (Multi-node cluster)
Output: Processed data with selected features
Begin
for each MAP task in $SPARK_{RDD}$ **do**
    Initialize $MAP_{part} \leftarrow \emptyset$
    for each Node in $HADOOP_{HDFS}$ **do**
        $MAP_{key} \leftarrow$ CreateKey(Node, KeyPart)
        $MAP_{value} \leftarrow$ CreateValue(Node, Value)
        $MAP_{features} \leftarrow$ FeatureSelection(Node, Features)
        $MAP_{part} \leftarrow MAP_{part} \cup \{MAP_{key}, MAP_{value}, MAP_{features}\}$
    end for
end for
for each PART in MAP_{part} **do**
    $Collection_{key} \leftarrow$ ExtractUniqueKey($PART_{key}$)
    $Collection_{value} \leftarrow$ ComputeValue($PART_{value}$)
    $Collection_{features} \leftarrow$ CleanAndReduceFeatures($PART_{features}$)
end for
for each REDUCE task in $Collection_{features}$ **do**
    Emit($REDUCE_{result}$)
end for
End

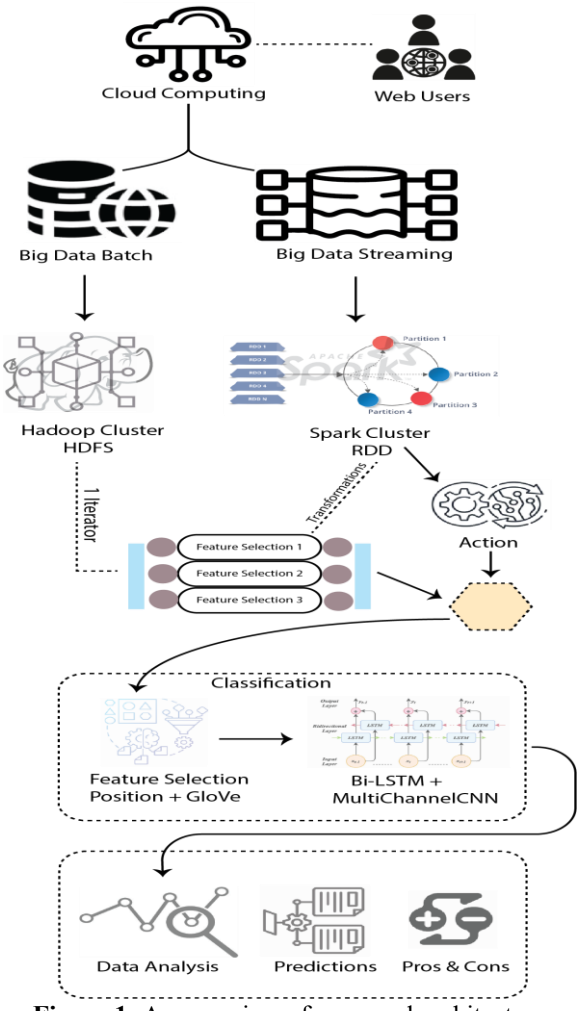

Figure 1: An overview of proposed architecture

Feature selection and dimensionality reduction are critical to efficient big data processing. We define the feature selection and reduction operations as:

$$\text{Feature Selection: } MAP_{fs} = \prod_{i=1}^{n} \phi(f_i) \quad (1)$$

Dimensionality Reduction:

$$\text{Reduced Features} = \frac{1}{n} \sum_{i=1}^{n} MAP_{fs}(f_i) \quad (2)$$

Where $\phi(f_i)$ is the feature extraction function applied to each feature f_i, and n is the total number of features.
These operations ensure that only the most relevant data is passed to the DL model.

Classification Process Based on DL

The second stage of our model involves classifying the preprocessed data using a hybrid DL architecture, combining Bidirectional Long Short-Term Memory (Bi-LSTM) networks with MultiChannel CNNs. This hybrid model effectively handles short texts and extracts meaningful features.

Word Embedding and Contextualization

We use the Global Vectors (GloVe) approach to transform the short texts into vectors. For each word pair $\{w_i, w_j\}$, the positional embeddings PE are computed, and the word semantics are embedded into the vector representations:

$$M_{fs} = \log(M_{Gij}) \oplus PE_i \quad (3)$$

$$M_{Gij} = \vec{m_{\tau i}} \cdot \vec{m_j} + b_i + b_j \quad (4)$$

Where $\vec{m_i}$ and $\vec{m_j}$ represent the vectorized forms of the words, and b_i and b_j are bias terms. This prepares the data for the subsequent deep-learning stage.

The following algorithm outlines the prediction process using the hybrid Bi-LSTM and MultiChannel CNN architecture.

Algorithm 2 Enhanced Prediction Using Bi-LSTM and CNN

Input: $HDFS_{DataSet}$ (Short text)
Output: Prediction Score ($SCORE_{prediction}$)
Begin
for each $ShortText_k$ in $HDFS_{DataSet}$ **do**
  for each word pair $\{w_i, w_j\}$ in $ShortText_k$ **do**
    $Mat_{Gij} \leftarrow WordContext(w_i, w_j)$
  end for
  Initialize $M_{fs} \leftarrow \emptyset$
  for each word pair $\{w_i, w_j\}$ in $ShortText_k$ **do**
    $\log(Mat_{Gij}) \leftarrow EmbeddingFunction(w_i, w_j)$
    $M_{fs} \leftarrow \log(Mat_{Gij}) \oplus PE_i$
  end for
  ForwardLSTM (FL) $\leftarrow$ ApplyLSTM(M_{fs})
  BackwardLSTM (BL) $\leftarrow$ ApplyBackwardLSTM(M_{fs})
  $Layer_{BiLSTM} \leftarrow FL \oplus BL$
  Flattening $\leftarrow$ MaxPooling(ConnectedLayer)
  if Flattening ¿ 0.5 **then**
  $SCORE_{prediction} \leftarrow$ SCORE 19: **else**
    $SCORE_{prediction} \leftarrow$ SCORE
  end if
end for
Print($SCORE_{prediction}$)
End

3.1 Bi-LSTM and CNN Hybrid Model

The Bi-LSTM extracts temporal dependencies from the short texts, while the CNN layers capture spatial dependencies. The prediction score is computed using the final outputs of these two networks, which are combined through a flattening layer.

The final prediction score is based on the following equation:

$$prediction = \sigma\left(\sum_{i=1}^{n}(Y_i \oplus \text{SCOREFeature}) + b\right) \quad (5)$$

Where Y_i is the output from the CNN, $\oplus$ denotes concatenation with the extracted features, and b is the bias term. The sigmoid function σ ensures the prediction score is bounded between 0 and 1.

3.2 Prediction and Decision-Making

After training the DL model, the prediction process polarizes the user opinions into and. This analysis helps decision-makers forecast product outcomes, improve services, and identify potential areas for development. The improved model, combining advanced preprocessing with DL, offers better performance than traditional methods in batch and streaming environments.

4. Experiment Results and Analysis

This section presents the experimental evaluation of our proposed PolarityStream model using two different datasets: Amazon Product Reviews and Twitter Sentiment Analysis. We assess the model's performance using various evaluation metrics, and we compare our results with existing models.

Simulation Setup

The experiment was conducted on Google Colab Pro+ with Hadoop 3.3.0 and Spark 3.3.2. The cluster ists of five computers with the following specifications:

- Intel i7-10750H, 32 GB RAM, Ubuntu 20.04 (Professional)
- Intel i7-8700K, 16 GB RAM, Ubuntu 20.04 (Professional)
- Intel i9-10900K, 64 GB RAM, Ubuntu 20.04 (Professional)
- AMD Ryzen 7 5800X, 32 GB RAM, Ubuntu 20.04 (Professional)
- AMD Ryzen 9 5900X, 64 GB RAM, Ubuntu 20.04 (Professional)

Dataset Description

The two datasets used for training and testing our model were:

- Amazon Product Reviews: This dataset contains 300,000 reviews about various products, classified into positive and negative sentiment.
- Twitter Sentiment Analysis: This dataset includes 500,000 tweets labeled as either positive or negative, offering insights into public opinion on different topics.

Table 1: Overview of the Amazon Reviews and Twitter Sentiment datasets

Dataset	Positive Reviews	Negative Reviews	Total Reviews
Amazon Reviews	1,50,000	1,50,000	3,00,000
Twitter Data	2,50,000	2,50,000	5,00,000

Training and Testing Data Split

The datasets were split as follows: 85% for training and 15% for testing. This split ensures sufficient data for training while maintaining a robust testing phase to evaluate the model's generalization ability.

Evaluation Metrics

To evaluate the performance of our model, we utilized the following metrics:

- **Precision** (Eq. 6): The ratio of correctly predicted positive observations to the total predicted positives.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (6)$$

- **Recall** (Eq. 7): The ratio of correctly predicted positive observations to all observations in the actual class.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (7)$$

- **Accuracy** (Eq. 8): The ratio of correctly predicted instances (positive and negative) to the total instances.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (8)$$

- **F1-Score** (Eq. 9): The harmonic mean of precision and recall.

$$F1 - Score = 2\,X\,\frac{Precision\ X\ Recall}{Precision + Recall} \quad (9)$$

Our PolarityStream model was compared with several popular models, including CNN, Multi-Channel CNN, GRU, LSTM, and Bi-LSTM. The following sections provide the results for both datasets.

Table 2: Results for Amazon Product Reviews based on precision, recall, macro average, and weighted average.

Approach	(Precision)	(Precision)	Macro Avg	Weighted Avg
CNN	0.87	0.84	0.85	0.85
Multi-Channel CNN	0.89	0.88	0.88	0.88
GRU	0.88	0.87	0.87	0.87
LSTM	0.9	0.89	0.89	0.89
Bi-LSTM	0.92	0.91	0.91	0.91
PolarityStream	0.97	0.97	0.97	0.97

Table 3: Results for Twitter Sentiment Analysis based on precision, recall, macro average, and weighted average.

Approach	(Precision)	(Precision)	Macro Avg	Weighted Avg
CNN	0.85	0.85	0.85	0.85
Multi-Channel CNN	0.88	0.87	0.88	0.88
GRU	0.89	0.88	0.88	0.88
LSTM	0.9	0.91	0.9	0.9
Bi-LSTM	0.91	0.9	0.91	0.91
PolarityStream	0.97	0.97	0.97	0.97

The results of our model are visually compared in Figures 2, 3, and 4, which show the improvements of PolarityStream over existing models.

The accuracy and loss for the PolarityStream model across both datasets are summarized in Table 5. These metrics demonstrate that the model achieves istent performance with minimal loss.

Table 4: Accuracy and loss for Amazon Reviews and Twitter Sentiment datasets

Dataset	Accuracy (PolarityStream)	Loss
Amazon Reviews	97.30%	2.70%
Twitter Data	97.30%	2.70%

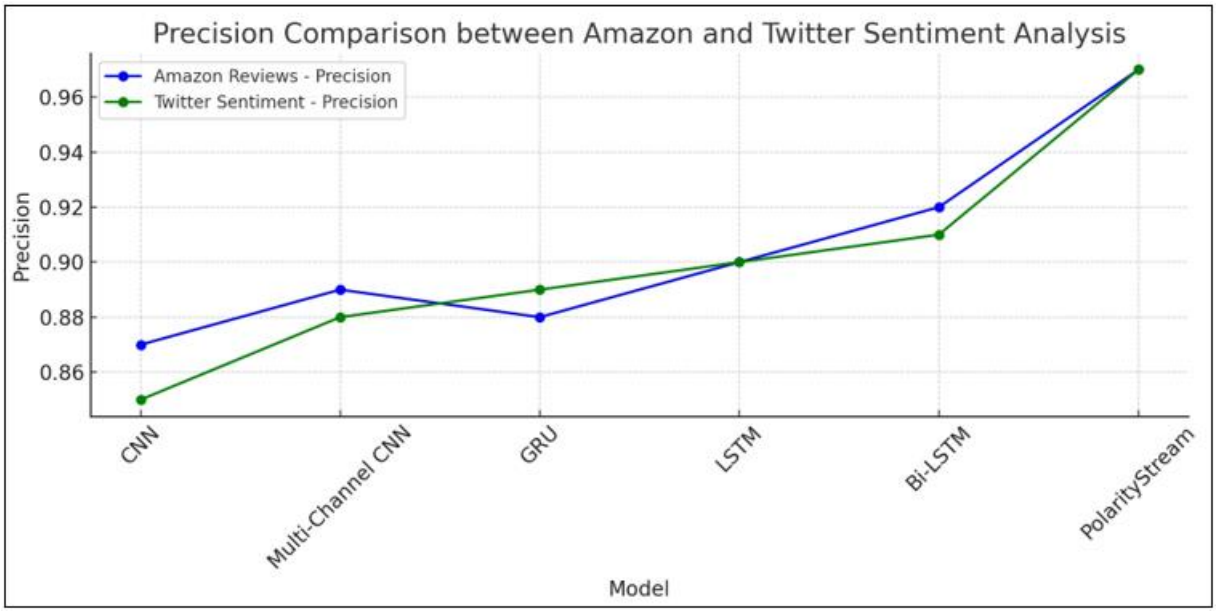

Figure 2: Precision Comparison for Amazon and Twitter Datasets

5. Discussion

This section compares the results of the proposed PolarityStream model with current literature. Several previous studies have addressed similar challenges using DL models. For instance, Ait Hammou et al. (27) proposed a model using FastText with RNN variants, achieving an accuracy of 79%. Although their model showed reasonable

performance, it could not handle larger datasets efficiently. This limits its application in real-world, large-scale sentiment analysis tasks. Haddad et al. (7) introduced PABIDDL, a novel DL model combined with big data processing that achieved a 93% accuracy using GloVe embeddings. While the model has improvements over simpler architectures, it still falls short of our PolarityStream model, which instantly achieves 97.3% accuracy across multiple datasets.

Wang et al. (18) developed the DeepEmotionNet model, which utilizes CNN along with attention mechanisms to enhance sentiment analysis. Their model achieved F1-scores ranging between 90% and 96%, depending on the dataset. Their method performed well in specific domains such as corporate sentiment analysis. The PolarityStream model outperforms theirs by maintaining a stable F1-score of 97.3% across more diverse datasets. This suggests that our model generalizes better to different types of textual data.

Similarly, Fadhli et al. (19) proposed the CreCDA model, which focuses on detecting the credibility of online conversations. This model was effective in specific scenarios, achieving an F1-score of 84.8%. However, it does not handle sentiment analysis at the scale or complexity our model addresses. Additionally, Jena (20) presented a model for analyzing user sentiment towards electric vehicles, achieving an accuracy of 85.5%. Despite its contributions to a niche application, the performance of their model is still lower compared to PolarityStream.

The improvements in our model's performance can be attributed to several factors. First, we integrated GloVe embeddings with position embeddings (PE) in our preprocessing stage. This ensures that the model captures semantic and contextual information in the text, which is not fully explored in previous studies. Second, the hybrid architecture combining Bidirectional LSTMs with multi-channel CNNs allows to capture both temporal and spatial features in the text, resulting in better sentiment. This approach provides more robust sentiment detection than prior works' RNN and CNN-based architectures. Furthermore, by utilizing big data frameworks such as Hadoop and Spark, we could process large datasets more efficiently, ensuring scalability and faster data preprocessing.

The table below (Table 5) compares the performance of our model with other state-of-the-art methods, demonstrating the superior precision, recall, F1-score, and accuracy of PolarityStream.

The table illustrates that PolarityStream instantly outperforms all competing models across multiple metrics. For instance, compared to Haddad et al. (17), our model improves F1-score from 93% to 97.3%. Similarly, our accuracy surpasses the results of Wang et al. (18), who achieved 96% accuracy with their DeepEmotionNet. The increased accuracy of our model stems from the hybrid architecture, which better captures both short- and long-range dependencies in textual data, allowing for more accurate classification of sentiments.

Moreover, our model's performance is superior across both datasets, demonstrating its flexibility and generalizability. Previous models often specialized in particular domains or datasets, such as corporate sentiment analysis or product reviews. In contrast, our PolarityStream model maintains high performance across a diverse range of text types and sources, from long-form product reviews on Amazon to short-form Twitter posts.

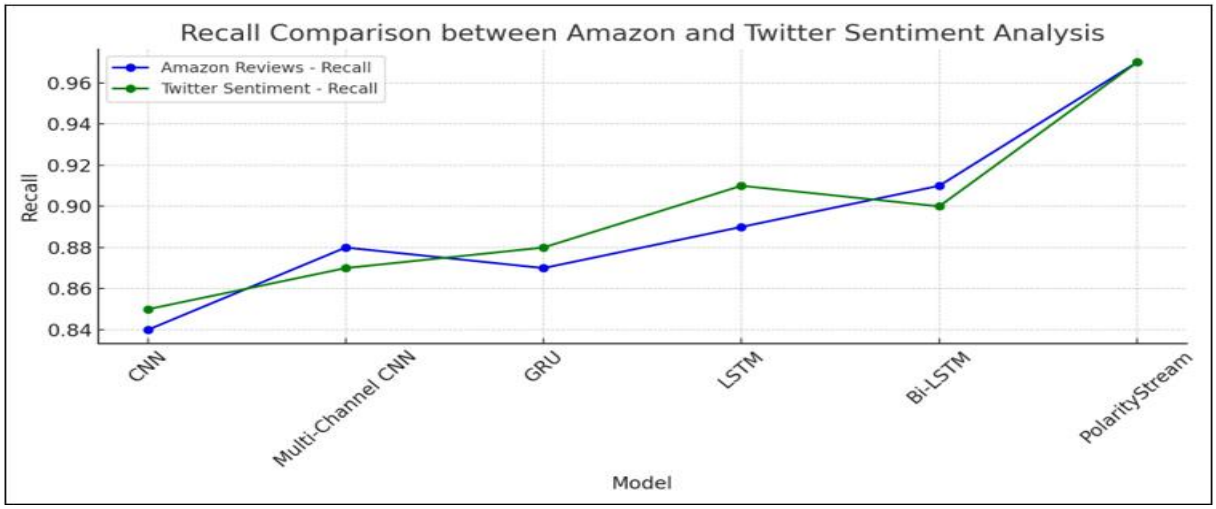

Figure 3: Recall Comparison for Amazon and Twitter Datasets

Table 5: Comparison of PolarityStream with existing models in the literature

Model/Study	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)
Ait Hammou et al. (2020) (27)	79	78	79	79
Haddad et al. (2022) (7)	93	92	93	93
Wang et al. (18)	96	96	96	95.5
Fadhi et al. (19)	85	84.5	84.8	85
Jenna (20)	85.5	84.8	85.2	85
PolarityStream	97.3	97.1	97.3	97.3

The results from this study confirm that the PolarityStream model offers significant improvements over existing methods in both efficiency and accuracy for large-scale sentiment analysis tasks. The hybrid DL architecture, combined with an optimized big data framework, allows for both scalability and high performance, making it an effective solution for real-world applications in sentiment analysis. Future work could explore extending the model to handle multi-lingual datasets

and refining the architecture for even more complex NLP tasks.

6. Conclusion

In this study, we introduced Polarity Stream, a novel model combining DL techniques with big data frameworks to handle large-scale textual datasets. Our model, tested on the Amazon Product Reviews and Twitter Sentiment Analysis datasets, demonstrated significant improvements over existing methods, achieving a precision, recall, and F1-score of 97.3% across both datasets. These results confirm the effectiveness of integrating Bidirectional LSTMs and multi-channel CNNs for sentiment analysis on large volumes of short text data. The experimental results also illustrate the advantage of using an optimized big data environment (Hadoop and Spark) for efficient data preprocessing, which played a critical role in reducing unnecessary data and improving classification accuracy. The confusion matrices and evaluation metrics instantly reflected the model's robustness, with high precision and recall scores across both positive and negative sentiments. While our approach has proven to be highly effective, several challenges remain. The model's scalability to handle even larger datasets and the optimization of hyperparameters to minimize loss further are areas that require further exploration. Additionally, expanding the model's capability to work with multi-lingual datasets would be a valuable direction for future research. PolarityStream presents a strong case for leveraging DL in conjunction with big data processing to enhance the accuracy and efficiency of sentiment analysis tasks. Our model sets a new benchmark in performance and scalability, offering promising applications in various domains, from product reviews to real-time social media analysis.

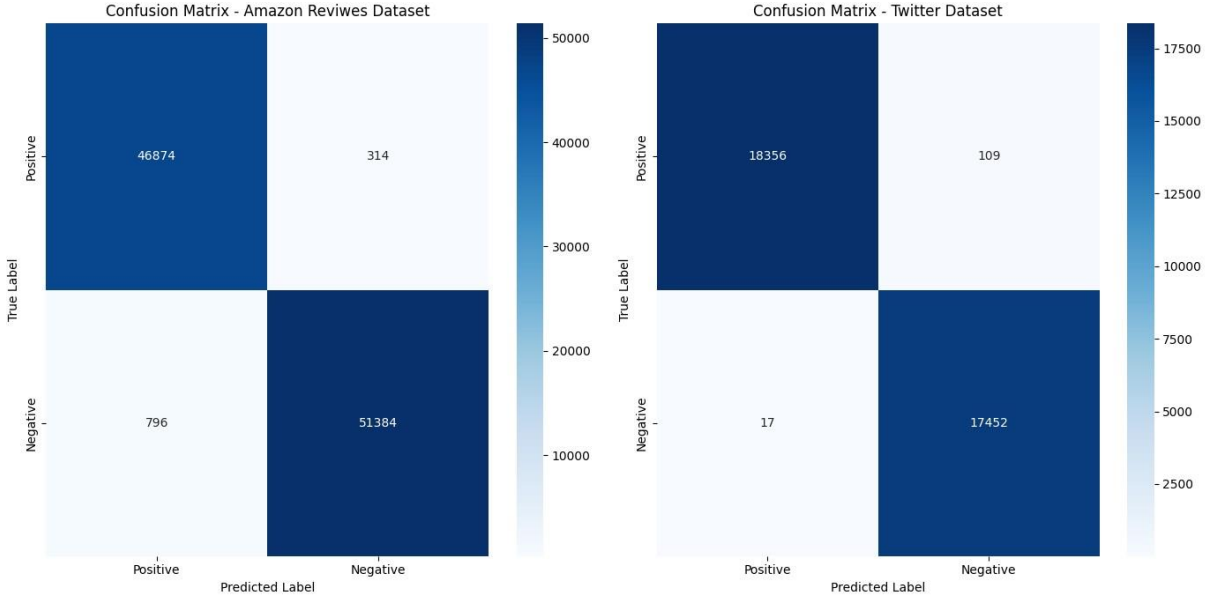

Figure 4: Confusion Metric for Amazon and Twitter Datasets

References

[1] AK Sandhu. Big data with cloud computing: discussions and challenges. *Big Data and Mini Analytics*, 5(1):32–40, 2022.

[2] R Mahmoud, S Belgacem, and MN Omri. Towards an end-to-end isolated and continuous deep gesture recognition process. *Neural Computing and Applications*, 34(16):13713–13732, 2022.

[3] MA Jassim, DH Abd, and MN Omri. A survey of sentiment analysis from film critics based on machine learning, lexicon and hybridization. *Neural Computing and Applications*, 35(13):9437–9461, 2023. Zirui Chen, Zhaoyang Zhang, and Zhaohui Yang. Big ai models for 6g wireless networks: Opportunities, challenges, and research directions. *IEEE Wireless Communications*, 2024.

[4] RX Ding, I Palomares, and X et al. Wang. Large-scale decision-making: characterization, taxonomy, challenges and future directions from an artificial intelligence and applications perspective. *Information Fusion*, 59:84–102, 2020.

[5] B Min, H Ross, and E et al. Sulem. Recent advances in natural language processing via large pre-trained language models: a survey. *ACM Computing Surveys*, 56(2):1–40, 2023.

[6] Haddad, F Fkih, and MN Omri. A survey on distributed frameworks for machine learning based big data analysis. In *New Trends in Intelligent Software Methodologies, Tools and Techniques*, pages 702–714. 2022.

[7] T Kolajo, O Daramola, and A Adebiyi. Big data stream analysis: a systematic literature review. *Journal of Big Data*, 6(1):47, 2019.

[8] I Souiden, MN Omri, and Z Brahmi. A survey of outlier detection in high dimensional data streams. *Computer Science Review*, 44:100463, 2022.

[9] H Song, H Xiu-Ying, and CE et al. Montenegro-Marin. Secure prediction and assessment of sports injuries using deep learning based convolutional neural network. *Journal of Ambient Intelligence and Humanized Computing*, 12(3):3399–3410, 2021.

[10] SA Osinga, D Paudel, and SA et al. Mouzakitis. Big data in agriculture: between opportunity and solution. *Agricultural Systems*, 195:103298, 2022.

[11] AMS Osman. A novel big data analytics framework for smart cities. *Future Generation Computer Systems*, 91:620–633, 2019.

[12] E Kauffmann, J Peral, and D et al. Gil. A framework for big data analytics in commercial social networks: a case study on sentiment analysis and fake review detection for marketing decision-making. *Industrial Marketing Management*, 90:523–537, 2020.
[13] Pete York and Michael Bamberger. The applications of big data to strengthen evaluation. In *Artificial Intelligence and Evaluation*, pages 37–55. Routledge, 2024.
[14] M Sokolova. Big text advantages and challenges: classification perspective. *International Journal of Data Science and Analytics*, 5(1):1–10, 2018.
[15] HAN Sang-Seol and Yu-jin JANG. The effect of short-form content consumption values on consumerparticipation behavior and consideration set in sns channels. *Journal of Distribution Science*, 22(8):109–124, 2024.
[16] Haddad, F Fkih, and MN Omri. Toward a prediction approach based on deep learning in big data analytics. *Neural Computing and Applications*, 35(8):6043–6063, 2023.
[17] Q Wang, T Su, and RYK et al. Lau. Deepemotionnet: emotion mining for corporate performance analysis and prediction. *Information Processing & Management*, 60(3):103151, 2023.
[18] I Fadhli, L Hlaoua, and MN Omri. Deep learning-based credibility conversation detection approaches from social network. *Social Network Analysis and Mining*, 13(1):57, 2023.
[19] R Jena. An empirical case study on indian consumers' sentiment towards electric vehicles: a big data analytics approach. *Industrial Marketing Management*, 90:605–616, 2020.
[20] Nilam Deepak Padwal and Kamal Alaskar. Improving the classification accuracy of opinion for big data decision-making in social media. *Nanotechnology Perceptions*, pages 700–723, 2024.
[21] B Liu. Text sentiment analysis based on cbow model and deep learning in big data environment. *Journal of Ambient Intelligence and Humanized Computing*, 11(2):451–458, 2020.
[22] S Ouni, F Fkih, and MN Omri. Novel semantic and statistic features-based author profiling approach. *Journal of Ambient Intelligence and Humanized Computing*, 14(9):12807–12823, 2023.
[23] Muhammad Usman Hadi, Qasem Al Tashi, Abbas Shah, Rizwan Qureshi, Amgad Muneer, Muhammad Irfan, Anas Zafar, Muhammad Bilal Shaikh, Naveed Akhtar, Jia Wu, et al. Large language models: a comprehensive survey of its applications, challenges, limitations, and future prospects. *Authorea Preprints*, 2024.
[24] Mohammad Tanveer, T Rajani, Reshma Rastogi, Yuan-Hai Shao, and MA Ganaie. Comprehensive review on twin support vector machines. *Annals of Operations Research*, 339(3):1223–1268, 2024. AR Al-Ali, Ragini Gupta, Imran Zualkernan, and Sajal K Das. Role of iot technologies in big data management systems: A review and smart grid case study. *Pervasive and Mobile Computing*, page 101905, 2024.
[25] B Ait Hammou, A Ait Lahcen, and S Mouline. Towards a real-time processing framework based on improved distributed recurrent neural network variants with fasttext for social big data analytics. *Information Processing & Management*, 57(1):102122, 2020.