International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 2, Mar-Apr-2026, ISSN (Online): 2395-566X

# Twitter Sentiment Analysis using Hybrid CNN-BiGRU Deep Learning Model

**Shubham Rathod, Dr.Rajeshwari Kannan**
Department of Computer Science Pimpri Chinchwad College of Engineering Pune

**Abstract - Social media platforms such as Twitter generate huge textual data every day, making sentiment analysis an important research area in Natural Language Processing (NLP). As there are many tweets that are noisy and reqired some contextual background . So the Research proposes the hybrid architecture of CNN and BiGRU so solve such issue. The Dataset which has been used is Sentiment 140 which contains 1.6 million entries.The methodology used is preprocessing , tokenization , CNN feature extraction and BiGRU contextual learning.The result shows the final accuracy using such architecture comes out to be 77.13%.Along with that there is use of flask web applicaton to create an interface.**

**Keywords - Twitter Sentiment Analysis, NLP, CNN, Bidirectional GRU, Sentiment140,Flask,Text Classification, Social Media Analytics**

## I. INTRODUCTION

The Social Media Platforms such as Twitter contains huge amount of data and is adding more to it , as the users are posting and commenting and give feedbacks etc. Analyzing such user-generated textual content provides valuable insights for businesses, researchers, and organizations.

Sentiment analysis is a Natural Language Processing (NLP) domain where the emotions is detected through the understanding the context behind the textual data .

Deep learning approaches have improved sentiment analysis performance by automatically learning feature representations from raw text. Convolutional Neural Networks (CNN) are helpful in extracting important local textual patterns, while recurrent neural networks such as Gated Recurrent Unit (GRU) are useful in capturing sequential contextual dependencies inside text sequences. Bidirectional GRU (BiGRU) helps in improving contextual text by processing textual sequences in both forward and backward directions.

This research proposes a hybrid CNN-BiGRU architecture for Twitter sentiment analysis. The proposed system combines CNN-based feature extraction with BiGRU-based contextual learning to make better sentiment classification performance. The Sentiment140 dataset was used for experimentation, and a Flask-based web application was developed for real-time sentiment prediction. The proposed model achieved an accuracy of 77.13% on the test dataset, showing effective sentiment classification capability for social media textual data.

## II. LITERATURE REVIEW

Sentiment analysis is one of the important research areas in Natural Language Processing (NLP) because of its rapid growth of social media platforms and online communication. Researchers have studied various machine learning and deep learning techniques for analyzing sentiments from textual data.

Traditional machine learning techniques such as Naive Bayes, Support Vector Machine (SVM), and Logistic Regression were initially used for sentiment classification tasks. Even though these methods achieved decent performance, they mainly depend on manual feature engineering and were not available to properly capture contextual relationships within textual sequences.

Deep learning approaches has improved sentiment analysis performance by automatically learning feature representations from raw text data. Convolutional Neural Networks (CNN) demonstrated strong capability in extracting local textual features and identifying important sentiment-bearing phrases from sentences.

During the building phase of this research, many deep learning models were tried and evaluated for Twitter sentiment analysis. Initially CNN-LSTM architectures

achieved approximately 50% accuracy because of the limited contextual learning ability optimization.Further experimentation using improved CNN-GRU and hybrid architectures has increased the classification accuracy to approximately 74%.
The proposed hybrid CNN-BiGRU model achieved the highest performance with a test accuracy of 77.13%. The combination of CNN-based feature extraction and BiGRU contextual understanding has improved sentiment classification capacity for Twitter textual data.
Recent updation in transformer-based architectures such as BERT and DistilBERT have shown high contextual understanding and state-of-the-art performance in NLP tasks. But, such transformer models want significantly higher computational resources and training complexity. Therefore, this research is based on a computationally efficient hybrid CNN-BiGRU architecture suitable for real-time sentiment prediction and lightweight deployment using Flask web applications.

## III. PROPOSED METHODOLOGY

The proposed system performs a hybrid deep learning architecture combining Convolutional Neural Network (CNN) and Bidirectional Gated Recurrent Unit (BiGRU) for Twitter sentiment classification. The workflow consists of data collection, preprocessing, tokenization, padding, deep learning model training, evaluation, and deployment using a Flask-based web application.

### A. Dataset Collection

The Sentiment140 dataset was used for research and model training. The dataset contains approximately 1.6 million Twitter tweets .
Dataset labels:

- 0 → Negative Sentiment
- 4 → Positive Sentiment

In order for better learning and efficient training, equal numbers of positive and negative tweets were selected during preprocessing.

### B. Data Preprocessing

Twitter data contains noisy textual information such as:

- URLs,
- hashtags,
- mentions,
- emojis,
- special symbols,
- abbreviations.

Preprocessing was done in order to improve model performance.
The preprocessing stage included:

1. Lowercasing text
2. Removing URLs
3. Removing special characters
4. Removing unnecessary symbols
5. Cleaning noisy textual content

Preprocessing helps for the text consistency and reduces irrelevant information while training.

### C. Tokenization

Deep learning models cannot process raw textual data directly. Therefore, first we need to convert contextual data into numerical token sequences using the Keras Tokenizer.
Example:

| Word | Token ID |
|---|---|
| happy | 15 |
| sad | 22 |

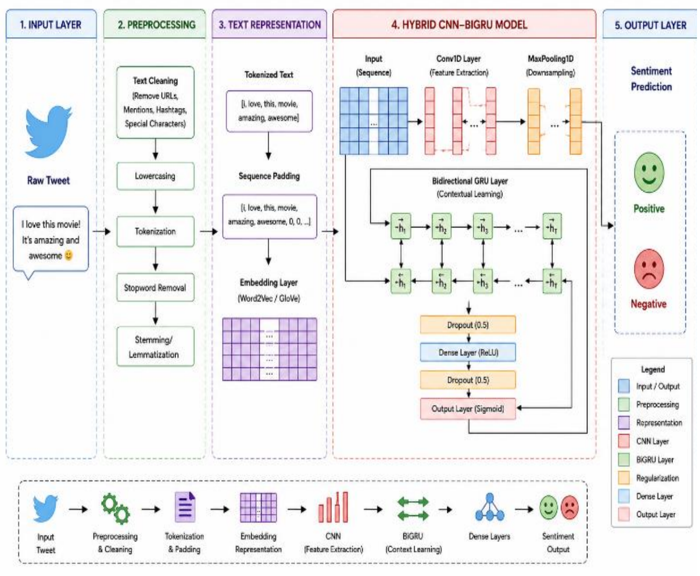
Sentence:
I am happy
becomes:
[4, 8, 15]

### D. Sequence Padding

Tweets have variable lengths, which creates inconsistency in neural network input dimensions. Therefore, sequence padding was used to make equal input length for all tweet sequences.
Example:
[15, 22]
becomes:
[0, 0, 0, 15, 22]
Padding helps in making better processing efficiency and model stability.

### E. Proposed Hybrid CNN-BiGRU Architecture

The proposed architecture joins CNN for feature extraction and BiGRU for contextual learning.




### 1) Embedding Layer

The embedding layer converts tokenized words into dense vector representations.
Mathematically:

$$E \in \mathbb{R}^{V \times d}$$

where:

- $V$represents vocabulary size,
- $d$represents embedding dimension.

The embedding layer captures semantic relationships between words.

### 2) Convolutional Neural Network (CNN)

CNN extracts the important local textual features and phrase-level sentiment patterns from tweets.
Convolution operation:

$$(f * g)(t) = \sum f(\tau) g(t - \tau)$$

CNN recognizes important patterns such as:

- "very happy"
- "hate this"
- "not good"

ReLU activation function comes after after convolution:

$$f(x) = \max(0, x)$$

### 3) MaxPooling Layer

MaxPooling decreases feature dimensionality but preserving important information.
Mathematically:

$$y_i = \max(x_i)$$

Advantages:

- reduces computational complexity,
- prevents overfitting,
- improves training efficiency.

### 4) Bidirectional GRU (BiGRU)

BiGRU is used to capture sequential contextual dependencies from both forward and backward directions.
GRU hidden state update equation:

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$$

BiGRU is actually used to improve contextual understanding of tweets and handles sequential dependencies effectively.

### 5) Dense Layer and Sigmoid Activation

The dense layer performs final sentiment classification.
Dense layer equation:

$$y = Wx + b$$

$$w$$

$$b$$

Sigmoid activation function changes output into probability values between 0 and 1.

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Interpretation:

- output closer to 1 → positive sentiment,
- output closer to 0 → negative sentiment.

### F. Loss Function and Optimization

Binary Cross Entropy loss function was used for model training.

$$L = -\frac{1}{N} \sum [y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})]$$

The Adam optimizer was used for good gradient-based optimization.

Weight updates were done using backpropagation:

$$\theta \leftarrow \theta - \eta \nabla J(\theta)$$

where:

- $\theta$= model parameters,
- $\eta$= learning rate,
- $\nabla J$= gradient of loss function.

### G. Flask-Based Deployment

A Flask web application has been developed for real-time sentiment prediction. Users can enter tweet text through the web interface, and the trained CNN-BiGRU model predicts the sentiment polarity quickly.
The deployment pipeline includes:

1. User input
2. Tokenization
3. Padding
4. Model prediction
5. Sentiment output display

The Flask-based deployment shows practical real-time applications of the given sentiment analysis system.

## IV. EXPERIMENTAL RESULTS AND DISCUSSION

The proposed hybrid CNN-BiGRU model was trained and checked using the Sentiment140 Twitter dataset. Experimental analysis was done to evaluate the effectiveness of different deep learning architectures for sentiment classification.

### A. Experimental Setup

The experiments were performed using Python-based deep learning libraries including TensorFlow, Keras, Scikit-learn, and Flask. Model training and evaluation were done on a local development environment using Visual Studio Code.

**Software and Tools Used**

| **Component** | **Technology** |
|---|---|
| Programming Language | Python |
| Deep Learning Framework | TensorFlow / Keras |
| NLP Processing | Keras Tokenizer |
| Development Environment | VS Code |
| Web Framework | Flask |
| Dataset | Sentiment140 |

### B. Training Configuration

The proposed CNN-BiGRU model was trained using the following hyperparameters:

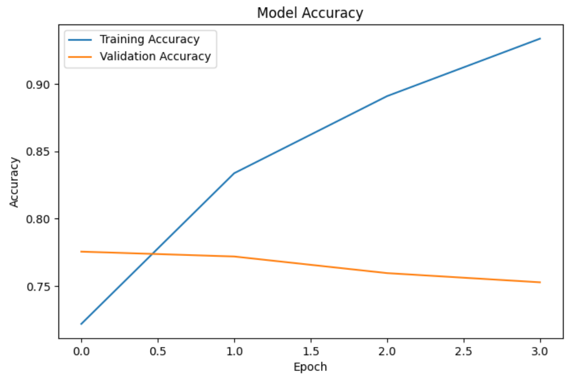



Fig. 3. Training and Validation Accuracy Graph

| **Parameter** | **Value** |
|---|---|
| Optimizer | Adam |
| Loss Function | Binary Cross Entropy |
| Batch Size | 128 |
| Epochs | 10 |
| Activation Function | ReLU, Sigmoid |
| Sequence Length | 100 |
| Embedding Dimension | 128 |

International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 2, Mar-Apr-2026, ISSN (Online): 2395-566X

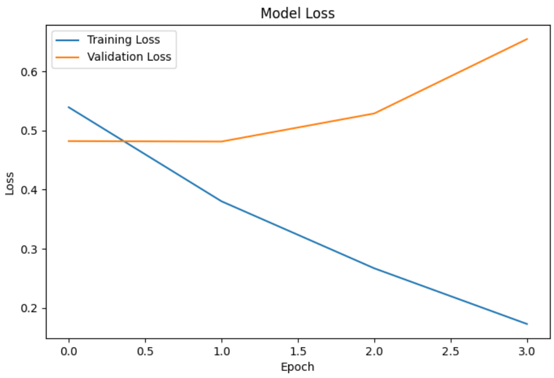



Fig. 4. Training and Validation Loss Graph

Early stopping was applied during training to reduce overfitting and improve generalization performance.

### C. Model Comparison

Multiple deep learning architectures were researched while experimentation to know the most effective model for Twitter sentiment analysis.

| Model | Accuracy |
|---|---|
| Initial CNN-LSTM Model | ~50% |
| Improved CNN-GRU Hybrid | ~74% |
| Proposed CNN-BiGRU Model | 77.13% |

The initial CNN-LSTM model had relatively lower accuracy due to limited contextual understanding and insufficient optimization. After that experimentation using improved hybrid architectures increased classification performance. The proposed CNN-BiGRU architecture got the highest test accuracy of 77.13%, showing improved contextual learning and feature extraction capability.

### D. Performance Analysis

The proposed hybrid architecture joins the advantages of CNN and BiGRU models.

**CNN Contribution**

CNN effectively extracts:

- local textual patterns,
- phrase-level features,
- sentiment-bearing keywords.

Examples:

- “very happy”
- “hate this”
- “not good”

**BiGRU Contribution**

BiGRU improves:

- contextual understanding,
- sequential dependency learning,
- bidirectional sentence interpretation.

This combination improves overall sentiment classification capability.

### E. Accuracy Evaluation

Model accuracy was calculated using:

$$\text{Accuracy} = \frac{\text{Correct Predictions}}{\text{Total Predictions}}$$

The final proposed model achieved:

| Metric | Value |
|---|---|
| Test Accuracy | 77.13% |
| Loss | 0.4780 |

The results shows that the given hybrid architecture effectively improves sentiment analysis performance for Twitter textual data.

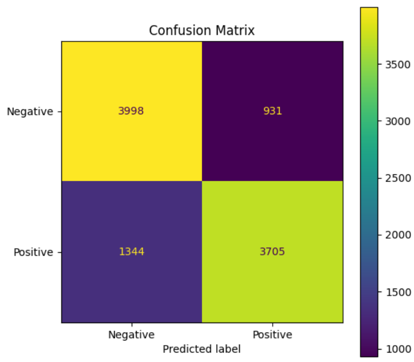



Fig. 5. Confusion Matrix of Proposed CNN-BiGRU Model

### F. Web Application Deployment

A Flask-based web application was developed in order to deploy the trained model for real-time sentiment prediction.
The deployment system allows users to:

1. Enter tweet text,
2. Process textual input,
3. Generate real-time sentiment prediction.

The web application shows the application of the given deep learning model in real-world sentiment analysis systems.

### G. Discussion

The experimental results shows that the hybrid deep learning architectures gives better sentiment analysis performance if we compared to standalone architectures. CNN improves feature extraction while BiGRU makes contextual understanding better.
Although transformer-based architectures such as BERT and DistilBERT may get higher accuracy, they require higher computational resources and training complexity. The proposed CNN-BiGRU model shows a good balance between performance and computational efficiency, making it good for lightweight deployment and real-time prediction systems.

## V. CONCLUSION AND FUTURE SCOPE

### A. Conclusion

This research proposed a hybrid CNN-BiGRU deep learning architecture for Twitter sentiment analysis using the Sentiment140 dataset. The proposed system is able to get andvantage of Convolutional Neural Network (CNN) for local feature extraction and Bidirectional Gated Recurrent Unit (BiGRU) for contextual sequence learning.
The preprocessing stage helps in better textual consistency through cleaning, tokenization, and sequence padding. CNN helps in extracting some of the important sentiment-bearing textual patterns, BiGRU gets contextual dependencies from both forward and backward directions, helping to improve the overall sentiment understanding.
Experimental analysis showed that the given hybrid architecture outperformed previously tested deep learning models. The final CNN-BiGRU model achieved a test accuracy of 77.13%, which shows the improvement in performance compared to initial CNN-LSTM and CNN-GRU experimental architectures.
The research generalizes that hybrid deep learning architectures can basically improve sentiment classification performance while maintaining computational efficiency and lightweight deployment capability.

### B. Future Scope

Although the proposed CNN-BiGRU model gets the sentiment classification performance, several improvements can be explored in future research.

**1. Transformer-Based Models**
Future work may include transformer-based architectures such as:

- BERT,
- DistilBERT,
- RoBERTa.

Transformer models provide better contextual understanding and state-of-the-art NLP performance.
Transformer attention mechanism:

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

Will be able to improve semantic understanding of textual data.

**2. Multiclass Emotion Classification**

The current system performs binary sentiment classification only:

- Positive,
- Negative.

Future systems may contain multiclass emotion detection such as:

- joy,
- anger,
- sadness,
- fear,
- surprise,
- love.

This would give deeper emotional analysis of social media text.

**3. Real-Time Twitter API Integration**

The proposed system can be extended using Twitter API integration for:

- live tweet analysis,
- trend monitoring,
- public opinion mining,
- brand sentiment tracking.

**4. Multilingual Sentiment Analysis**

Future research can be focused on multilingual sentiment analysis using:

- regional languages,
- code-mixed text,
- multilingual transformer models.

This would help to improve the application across social media platforms.

**5. Explainable AI (XAI)**

Future work may have Explainable AI techniques to improve deep learning predictions and increase the user trust in sentiment analysis systems.

**References**

[1] Go, A., Bhayani, R., and Huang, L., “Twitter Sentiment Classification using Distant Supervision,” Stanford University, 2009.

[2] Kim, Y., “Convolutional Neural Networks for Sentence Classification,” EMNLP, 2014.

[3] Cho, K. et al., “Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation,” EMNLP, 2014.

[4] Hochreiter, S. and Schmidhuber, J., “Long Short-Term Memory,” Neural Computation, 1997.

[5] Devlin, J. et al., “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” NAACL, 2019.

[6] Vaswani, A. et al., “Attention is All You Need,” NeurIPS, 2017.

[7] Dang, N. et al., “Hybrid Deep Learning Models for Sentiment Analysis,” Complexity Journal, 2021.

[8] Comparative Analysis of CNN and LSTM Neural Networks for Sentiment Classification on the Sentiment140 Dataset, IJISRT, 2024.

[9] Sentiment Analysis on Twitter Data using Deep Learning Approach, ResearchGate Publication.

[10] Efficient Twitter Sentiment Classification using Subjective Distant Supervision, arXiv Publication.