Sentiment Analysis on Bengali Text

Hasin Md Daiyan
dept. of CSE
AUST
Dhaka, Bd.
hasin.cse.20200204055@aust.edu

Ahmed Al Nahian
dept. of CSE
AUST
Dhaka, Bd.
ahmed.cse.20200204063@aust.edu

Amit Karmakar
dept. of CSE
AUST
Dhaka, Bd
amit.cse.20200204056@aust.edu

***Abstract*—Sentiment analysis, the task of automatically identifying the emotional polarity expressed in text, has gained significant importance in recent years due to the massive growth of user-generated content on the internet and social media platforms. This work tackles the problem of accurate sentiment classification of Bengali text using deep learning approaches.**

***Index Terms*—CNN, Bag of Words, Tf-Idf,**

I. Introduction

Sentiment analysis is the area which deals with judgments, responses as well as feelings, which is generated from texts.[1] When more computing power and a large amount of training data are available, the deep-learning-based approach to sentiment analysis classification can give more accurate results than the traditional sentiment classification methods that use SVM, NB, etc.[2]

II. Motivation

Kim [3] demonstrated the effectiveness of a simple convolutional neural network (CNN) architecture with one convolutional layer applied to pre-trained word vectors from an unsupervised neural language model. The word vectors were trained on a massive 100 billion word corpus from Google News. Remarkably, this straightforward CNN model with minimal hyperparameter tuning and static word embeddings achieved state-of-the-art performance across multiple sentiment analysis benchmarks for English text. Inspired by Kim's work showcasing the power of deep learning models, particularly CNNs, for sentiment analysis tasks in resource-rich languages like English, we are motivated to explore similar deep learning-based approaches for the low-resource Bengali language. Despite the vast number of native Bengali speakers, there is a dearth of resources and research dedicated to developing effective sentiment analysis tools for this language.

III. Literature Review

In a study by Haq, Haque, and Uddin [4], they discovered that by combining the Word2Vec embedding layer with a fusion of CNN-LSTM architecture, they were able to detect emotions from raw textual data with an impressive accuracy of 90.49 percent. Similarly, in another research endeavor [5], the highest accuracy achieved was 88.59 percent using the CNN-BiLSTM hybrid model, leveraging GloVe feature vectors. In yet another study [6], it was revealed that among CNN, LSTM, and BiLSTM models, the CNN with GloVe word embedding outperformed the others, achieving an accuracy of 94.57 percent.These studies demonstrate the effectiveness of various deep neural network architectures.

Identify applicable funding agency here. If none, delete this.

IV. Methodology

A. Data Collection and Preprocessing

For this study, a dataset of Bengali text comments was collected, along with their corresponding sentiment labels . The text data underwent several preprocessing steps to clean and prepare it for analysis. First, non-Bengali characters were removed from the text using regular expressions. Next, the text was tokenized into individual word tokens using the BasicTokenizer from the bnlp library [7]. Stopwords were then removed from the tokenized text using a predefined list of Bengali stopwords. Finally, the tokens were stemmed using the BanglaStemmer from the bangla stemmer library [8] to obtain the root forms of words.

B. Feature Extraction

The preprocessed text data was converted into numerical feature vectors suitable for input to deep learning models. Two different feature extraction techniques were employed:

- Term Frequency-Inverse Document Frequency (TF-IDF): TF-IDF vectors were generated using the TfidfVectorizer from the scikit-learn library , representing the importance of each word in the corpus.
- Bag-of-Words (BoW): BoW vectors were generated using the CountVectorizer from scikit-learn , capturing the presence or absence of words in the corpus.

C. Model Development

Three deep learning models were used and evaluated for sentiment analysis of Bengali text:

- Convolutional Neural Network (CNN): A CNN model was constructed with one convolutional layer followed by global max pooling, dense layers, and a dropout layer.

TABLE I
RESULT

	Model		
Logistic Regression	***CNN***	***RNN***	***LSTM***
Acc: 0.4174	Acc: 0.3107	Acc: 0.2718	Acc: 0.1650

The CNN aimed to capture local patterns and features in the text data.

- Recurrent Neural Network (RNN) with Simple RNN: An RNN model was built using the SimpleRNN layer, followed by dense layers and a dropout layer. The RNN aimed to model sequential dependencies in the text data.
- Long Short-Term Memory (LSTM): An LSTM model was constructed, similar to the RNN model but using the LSTM layer instead of SimpleRNN. LSTM networks are capable of capturing long-range dependencies in sequential data.

D. Model Training and Evaluation

The dataset was split into training and testing sets, with 90 percent of the data used for training and 10 percent for testing. The labels were encoded using a LabelEncoder from scikit-learn [3] and converted to one-hot encoding for multi-class classification. The TF-IDF and BoW vectors were reshaped to match the input requirements of the deep learning models. Each model (CNN, RNN, and LSTM) was compiled with categorical cross-entropy loss, Adam optimizer, and accuracy metric. The models were trained on the training data for a specified number of epochs (e.g., 10 epochs) and batch size (e.g., 32). The performance of each model was evaluated on the testing data, measuring the loss and accuracy metrics.

V. RESULT AND ANALYSIS

CNN model achieved a test accuracy of 0.3107, which is lower than the logistic regression model's accuracy. However, the training loss and accuracy values suggest that the model may not have converged properly or could be underfitting the data.Recurrent Neural Network (RNN) with Simple RNN: The RNN model with Simple RNN layers achieved a test accuracy of 0.2718, which is lower than both the logistic regression and CNN models. The training loss and accuracy values indicate that the model is not fitting the data well.

The LSTM model encountered a numerical issue during training, resulting in NaN (Not a Number) values for the test loss. This could be due to various reasons, such as numerical instability, poorly initialized weights, or inappropriate hyperparameter settings. Overall, the logistic regression model performed better than the deep learning models (CNN, RNN, and LSTM) in terms of accuracy for sentiment analysis on the Bengali text data. However, the performance of all models is relatively low, indicating the need for further improvements and adjustments.

CONCLUSION

This study explored the application of deep learning models, including Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Long Short-Term Memory (LSTM) networks, for sentiment analysis of Bengali text. The traditional machine learning approach using logistic regression outperformed the deep learning models, achieving an accuracy of 0.4174 compared to 0.3107 (CNN), 0.2718 (RNN), and 0.1650 (LSTM). The suboptimal performance of the deep learning models may be attributed to factors such as limited training data, the complexity of the Bengali language, and the need for further architectural and hyperparameter tuning. The LSTM model encountered numerical instability during training, highlighting the challenges associated with applying these techniques to low-resource languages like Bengali. While room for improvement exists, this study shows importance sentiment analysis tools for the Bengali language,

REFERENCES

[1] JKoyel Chakraborty, Siddhartha Bhattacharyya, Rajib Bag, Aboul Alla Hassanien, Sentiment Analysis on a Set of Movie Reviews Using Deep Learning Techniques

[2] Kamal Sarkar* Sentiment Polarity Detection in Bengali Tweets Using Deep Convolutional Neural Networks

[3] Y. Kim, Convolutional neural networks for sentence classification, in: Proceedings of the 2014 Conference on EMNLP, pp 1746–1751, ACL, Doha, Qatar, 2014.

[4] Hoq, M., Haque, P., Uddin, M.N. (2021). Sentiment Analysis of Bangla Language Using Deep Learning Approaches..

[5] S. Islam, M. Jahidul Islam, M. Mahadi Hasan, S. M. Shahnewaz Mahmud Ayon and S. Shabnam Hasan, "Bengali Social Media Post Sentiment Analysis using Deep Learning and BERT Model

[6] M. Rabeya, M. R. Tuly, M. S. Mahmud and A. Sattar, "Sentiment Analysis of Bengali Textual Comments in Field of Sports Using Deep Learning Approach August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] bnlp: Bengali Natural Language Processing Toolkit. https://github.com/bnlp/bnlp-toolkit

[8] bangla-stemmer: A Rule-Based Stemmer for Bengali. https://github.com/bnlp/bangla-stemmer