Fake News Detection Based on Deep Learning

1st Ashfia Jannat Keya
Department of Computer Science and Engineering, Bangladesh University of Business and Technology
Dhaka-1216, Bangladesh
ashfia@bubt.edu.bd

2nd Shahid Afridi
Department of Computer Science and Engineering, Bangladesh University of Business and Technology
Dhaka-1216, Bangladesh
shahidafridi.cse36@gmail.com

3rd Afroza Siddique Maria
Department of Computer Science and Engineering, Bangladesh University of Business and Technology
Dhaka-1216, Bangladesh
afrozasiddiquemaria@gmail.com

4th Snaha Sadhu Pinki
Department of Computer Science and Engineering, Bangladesh University of Business and Technology
Dhaka-1216, Bangladesh
snahasadhu524@gmail.com

5th Joy Ghosh
Department of Computer Science and Engineering, Bangladesh University of Business and Technology
Dhaka-1216, Bangladesh
joyghosh7272@gmail.com

6th M. F. Mridha
Department of Computer Science and Engineering, Bangladesh University of Business and Technology
Dhaka-1216, Bangladesh
firoz@bubt.edu.bd

2021 International Conference on Science & Contemporary Technologies (ICSCT) | 978-1-6654-2132-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICSCT53883.2021.9642565

***Abstract*—Fake news is invalid and misleading information that is conveyed as accurate news. Fake news detection has become indispensable in modern society because of the extreme propagation of false news on social platforms and news portals. Several studies have been released that use fake news on social platforms instead of news content for decision-making. Therefore, this paper introduces an automated model for detecting fake news relying on Deep Learning (DL) and Natural Language Processing (NLP) for a low-resource language like Bangla, utilizing news content and headline features. We propose an ensemble approach of Convolutional Neural Network (CNN) and Gated Recurrent Unit (GRU) with a pre-trained GloVe embedding method that achieved an accuracy of 98.71% on the test data. For comparison, the combination of Long short-term memory (LSTM) and CNN with GloVe is trained using the same dataset and parameters. We also experimented on a benchmark dataset containing English news with our suggested model and achieved an accuracy of 98.94%. Our model's performance is evaluated using diverse evaluation metrics, including accuracy, recall, precision, f1-score, etc.**

***Index Terms*—Fake News, Natural Language Processing, Deep Learning, Ensemble Approach, CNN+GRU.**

I. Introduction

Fake news is a terrific social issue. Few organizations and systems work together to distribute fake news around the world. People sometimes spread fake news intentionally to defame someone else and sometimes spread without any intention or knowing the actual story. Believing others opinions without judging and lack of knowledge causes the dissemination of misleading information in society. According to research, the spread of fake news on social media has a long-lasting effect on those who are not as wise. And holds them back from making the proper judgments [1]. With the expansion of the internet and social media, this fake news spreading rate is increasing drastically.

Many people already have access to social media, but they have no idea what they are reading or posting. People usually follow celebrities, religious or political figures, and believe or support them. Therefore, if those leaders post any news without first verifying the truth, false information spreads quickly. Even fake news has significant regional impacts. Disruption of the American presidential election is an excellent example of fake news, and distorted views of people [2].

Several scientific communities' efforts are seen before, in which they attempt to discover how to detect fake news from different perspectives (propagation patterns, source authenticity, writing style) [3]. Researchers have suggested various strategies to prevent false information. Diverse Machine Learning (ML) algorithms and Neural networks are applied for the detection of fake news [4]. Recently utilizing the Ensemble approach, the performance of the model is greatly increased [5]. So far, this sort of work has only been performed on English news. Currently, around 265 million people speak Bengali, as it is the 7th language in the world in terms of speakers. But there are few works done to tackle the risk of fake news written in Bengali [6]. The Overall contribution of the paper includes:

- We introduce an ensemble approach with a pre-trained word embedding method, GloVe, for the Bengali fake news detection. This is the first research in fake news detection that proposes a combination of CNN and GRU to the best of our knowledge.
- We investigate models performance with an established Bengali dataset, "BanFakeNews," and an English dataset named "Fake News" utilizing NLP applications.
- We perform several experiments with the Bengali dataset and trained CNN and LSTM ensemble model following the same process to compare with our proposed model.

The paper is structured as the following: Section II highlights the related work in this domain. Section III illustrates the proposed methodology. Section IV depicts the evalauation metrics. We discuss the results of our experiments in section V. Finally, Section VI brings the paper to a conclusion.

II. Related Work

Fake news and rumours are constantly growing through the internet. Several researchers are working to solve this problem. We have discussed the related works based on three approaches: 1) Machine Learning Approach, 2) Deep Learning Approach, and 3) Ensemble Approach.

A. Machine Learning Approach

Ibrishimova and Li [7] proposed a hybrid framework that combines 5 NLP features with three features of knowledge verification. The authors used logistic regression as their classifier and trained using Kaggle Fake News Dataset. Granik and Mesyura [8] utilized a dataset of Facebook news contents and proposed a system for detecting fake news based on the Naive Bayes classifier. However, the proposed model's performance was lacking, as it only obtained a 74% accuracy. Hussain et al. [6] used Multinomial Naive Bayes (MNB) and Support Vector Machine (SVM) classifiers for identifying Bangla fake news. The research concludes that SVM, combined with the linear kernel, has an accuracy of 96.64%, which is marginally better than MNB's 93.32% accuracy on the proposed dataset.

B. Deep Learning Approach

Bahad et al. [9] proposed a Bi-directional LSTM approach to detect fake news. They compared the Bi-directional LSTM model's accuracy to that of CNN, vanilla RNN, and unidirectional LSTM. It was found that the bi-directional LSTM model outperforms all other models. Girgis et al. [10] used Vanilla RNN, GRU, and LSTM on the LIAR dataset. They observed that the GRU reached 21.7% test accuracy, which is the best of their results. The authors plan to use a hybrid model that incorporates CNN and GRU techniques on the LIAR dataset. Another study by Kaliyar et al. [11] proposed a deep CNN (FNDNet) that can automatically learn the biased features for fake news classification through multiple hidden layers. Their model obtained an accuracy of 98.36%. The authors plan to utilize a multimodel-based approach combining the pre-trained word embeddings to improve their work.

C. Ensemble Approach

A study by Sangamnerkar et al. [12] performed various ensemble techniques with ML classifiers. An ensemble of Logistic Regression (LR), Decision Tree (DT), and Bagging Classifiers combined with a hard-voting ensemble technique produces the best results, with over 88% accuracy. Hakak et al. [13] recommended an ensemble model for fake news detection comprising three ML classifiers which are Random Forest, Decision Tree, and Extra Tree Classifier. The proposed model performed better in the ISOT dataset with an accuracy of 100%, but on the Liar dataset, it obtained only 44.15% accuracy. Agarwal and Dixit [14] introduced a method that includes an ensemble network for determining how news stories, writers, and titles are represented simultaneously. They tested various machine learning algorithms for higher accuracy, including CNN, KNN, SVM, LSTM, and Naive Bayes, and noticed that LSTM have the best accuracy with 97%. Asghar et al. [15] proposed a system for fake news detection, combining Bidirectional LSTM with CNN for classifying the tweet into rumours and non-rumours. Though their model achieved an accuracy of 86.12%, they believe that by combining diverse features, they can get more reliable results. A study by K. Shu et al. [16] wants to improve their work by including features such as the article's source or author, as well as user feedback.

Previous researchers focused on combining CNN, and LSTM, whereas GRU is structurally and functionally similar to LSTM. Still, GRU is simpler, more powerful, and uses less memory since it has only two gates, namely reset and update. The strength of LSTM in learning long-term dependencies is retained in GRU. Thus, we propose an ensemble approach of CNN and GRU utilizing news content and headline features.

III. Proposed Methodology

The architecture of our proposed methodology is shown in Fig. 1. A further detailed description of our methodology is given in the following sections.

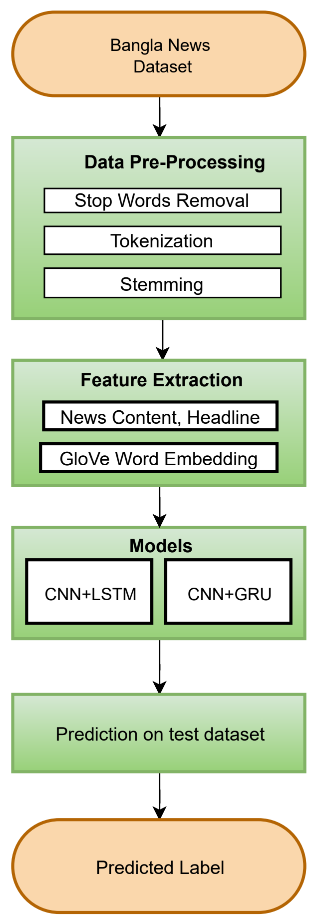


Fig. 1: Architecture of our methodology.

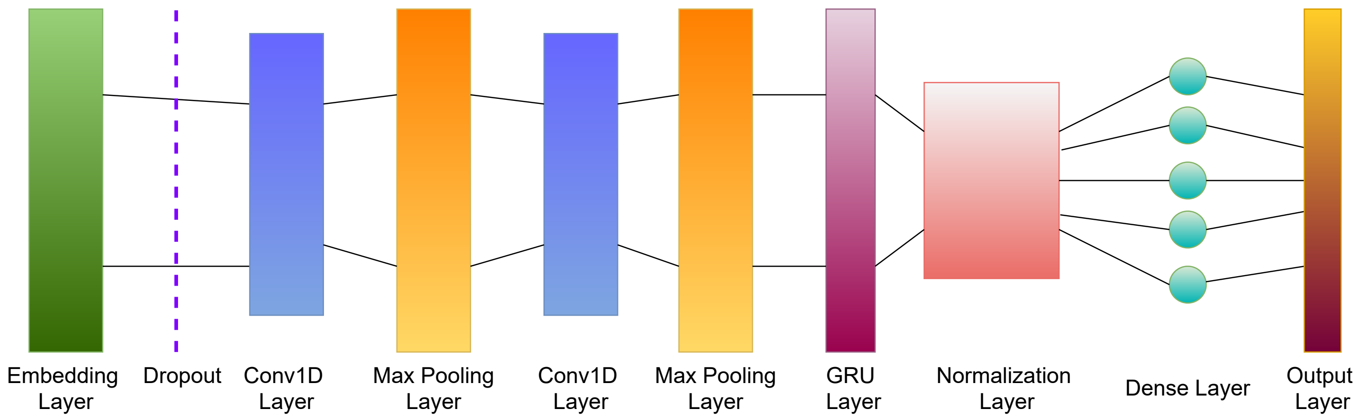


Fig. 2: Proposed model.

A. Dataset Description

We have used the "BanFakeNews" dataset [1] and the "Fake News" dataset [2] that are publicly available and collected from Kaggle. The "BanFakeNews" dataset [17] consists of approximately 50K news in Bengali, and the "Fake News" dataset consists of 20,800 news in English. The datasets are provided as a CSV file. There are seven attributes in the "BanFakeNews" that are: articleID, domain, date, category, headline, content, and label. And the "Fake News" dataset has five attributes which are: id, title, author, text, and label. The Table I displays the datasets details.

TABLE I: Description of the Datasets

Dataset	***True News***	***Fake News***	***Total***
BanFakeNews	48,678	1299	49,977
Fake News	10,413	10,387	20,800

B. Data Pre-processing

Data pre-processing is essential to reduce noisy and unnecessary data. For Bengali news, we merged two files consists of true news and fake news from the "BanFakeNews" dataset. For English news, we take the pre merged "Fake News" dataset. We considered News content and headline features for the "BanFakeNews" dataset. Author, text, and title features are considered for the "Fake News" dataset. NLP methods, including stemming, tokenization, and stop words removal, are used to convert the raw data with Keras and TensorFlow libraries' help. Stopwords are words that often appear in our collected data but have no meaning concerning features. Thus, we discovered all stopwords in the "BanFakeNews" dataset, e.g. `ই,' `|,' `আমরা,' `কই', `তখন,' and shorten execution time and saved memory space by eliminating these stopwords. For "Fake News" dataset, 'again,' 'do,' 'off,' 'with' are removed. Next, we identified words with similar meanings and used stemming with NLTK's porter stemmer. Finally, tokenization is applied to break all headlines into a word vector from our merged texts. After tokenizing, 72,355 and 2,47,340 unique tokens are found for Bengali and English datasets, respectively. Numerical sequences are added, replacing textual sequences and padded to the highest sequence length of 1000. For training, testing, and validation, we divided the dataset into a ratio of 80:10:10.

C. Feature Extraction

We removed some trivial features to construct a new collection of features to minimize the dimension of our dataset. The final sample has fewer features than the original dataset. The benefit of eliminating features is that the computational time is decreased, resulting in a more remarkable performance. Most algorithms run much faster if there are fewer dimensions to consider. Following the data pre-processing steps, we have removed articleID, date, category, and domain features from the "BanFakeNews" dataset. Cause these features are in the English Language while others are in Bengali. News content and headline features are considered in the "BanFakeNews" dataset. Author, text, and title features are considered in the "Fake News" dataset for further process.

We have applied pre-trained word embedding for our model. It is a form of transfer learning and is trained on large datasets. After the feature extraction, word embedding is used to mapping a term to a list of vectors. We combined pre-trained word embedding with our ensemble model. Global Vectors for Word Representation (GloVe) is used after feature extraction to map words to a list of vectors. An unsupervised learning algorithm, GloVe, is an approach to create word embeddings. It is easier to train over larger data as parallel implementation can be performed in gloVe. In this work, we have used the "bn_glove.39M.zip" and "glove.6B.zip" for Bengali and English datasets, respectively. We prepared the word embedding on our Bengali and English dataset that consists of 1,78,153 and 4,00,000 words, respectively. From several embedding vector sizes, we selected the 100-dimensional version. GloVe aims to emphasize the vectors of a word in the vector space for achieving sub-linear relationships.

[1]https://www.kaggle.com/cryptexcode/banfakenews
[2]https://www.kaggle.com/c/fake-news/data

The GloVe provides cheaper weight toward ubiquitous word sets for limiting the unnecessary stop words that do not control the training progress [11].

D. Model

1) Embedding and Dropout Layer: The embedding layer is the model's first layer, which takes input features and transforms every word into a 100-dimensional vector. If any text contains less than the maximum number of tokens, it is padded to equal length. The dropout layer receives these word vectors. We have added a dropout layer with a 0.2 dropout rate for our regularization technique, which means input values smaller than the dropout rate are dropped.

2) Convolution and Max-Pooling Layer: We have applied simple CNN with two convolution blocks, each consisting of a single Conv-1D and Max Pooling layer. We have used 32 filters with kernel size 5 in the first layer, and for the second layer, we have used 64 filters with kernel size 3. Each filter identifies more than one feature in the text with the help of the ReLu activation function. Then to the output of each CNN neuron, the ReLu activation function is used. This activation function converts any negative value to zero. The values are then fed to the 1-D Max-pooling layer. The Max-Pooling layer reduces dimensionality by conserving the learn patterns. The pool size is set to 2.

3) GRU Layer: After the convolution blocks, a GRU layer is used. We have used GRU as it helps to solve our problems of gradient vanishing and explode gradient. It also helps to handle long sequential textual data over its recurrent architecture. Since GRU has fewer tensor operations, it is easier to train than LSTM. We trained our proposed model with 100 epochs. And 20% dropout, 20% recurrent dropout at the GRU layer. Rest, we have kept default values for better accuracy.

4) Batch Normalization and Dense Layer: We applied Batch Normalization (BN) for standardizing the inputs to the dense layer. Standardizing the inputs indicates that inputs to the dense layer should have approximately zero mean and unit variance. The wholly connected dense layer is the final layer of our proposed model. It generates a single output. A Softmax activation function follows this layer. Small batches of 128 have been used to train and evaluate the proposed model depicted in Fig. 2.

IV. Performance Evaluation Metrics

Several performance measures are taken into consideration to evaluate our model. The Confusion Matrix, Accuracy (A), Precision (P), Recall (R), F1-score (F), and ROC curve are used as evaluation metrics for the proposed model.

1) Confusion Matrix: The confusion matrix shows an overview of model performance on the testing dataset from the known true values. It gives us a review of the model's success and useful results of true positive, true negative, false positive, and false-negative. Our model's confusion matrix with 2,080 instances of the test set of the "Fake News" dataset is given in Fig. 3.

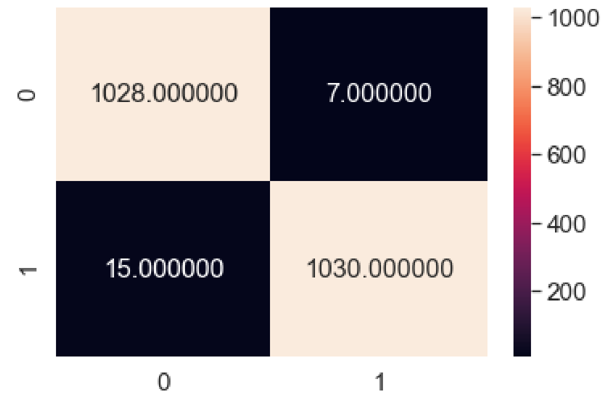


Fig. 3: The confusion matrix of the CNN+GRU model.

2) Accuracy: Accuracy Score, also known as classification accuracy rating, is determined as the percentage of correct predictions to total predictions made by the model. We depicted the accuracy (A) as given the formula in equation (1).

$$A = \frac{TruePositive + TrueNegative}{TotalNumberofPredictions} \tag{1}$$

3) Precision: When the number of true positive results is divided by the total number of positive results, including those that were incorrectly identified, it is known as precision (P). Precision is computed using equation (2).

$$P = \frac{TruePositive}{Positive + FalsePositive} \tag{2}$$

4) Recall: When the total number of samples that should have been identified as positive is used to divide, the number of true positive results is referred to as recall (R). The recall is computed using equation (3).

$$R = \frac{TruePositive}{TruePositive + FalseNegative} \tag{3}$$

5) F1-score: The accuracy of the model for each class is defined by the F1-score (F1). If the dataset is not balanced, then the F1-score metric is usually used. To show the proposed model's performance, we have used F1-score as an evaluation metric. F1-score computation is done using the following equation (4).

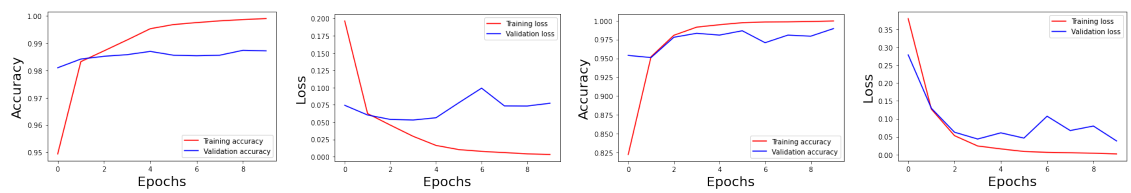
$$F1 = 2 \times \frac{precision \times recall}{precision + recall} \tag{4}$$

6) ROC curve and AUC: The Receiver Operating Characteristics (ROC) curve is used to show the success of a classification model across several classification thresholds. True Positive Rate (Recall) and False Positive Rate (FPR) are used in this curve. AUC is an abbreviation for "Area Under the ROC curve." In other words, AUC tests the whole two-dimensional field under the entire ROC curve. The FPR is defined as in equation (5).

$$FPR = \frac{FalsePositive}{FalsePositive + TrueNegative} \tag{5}$$

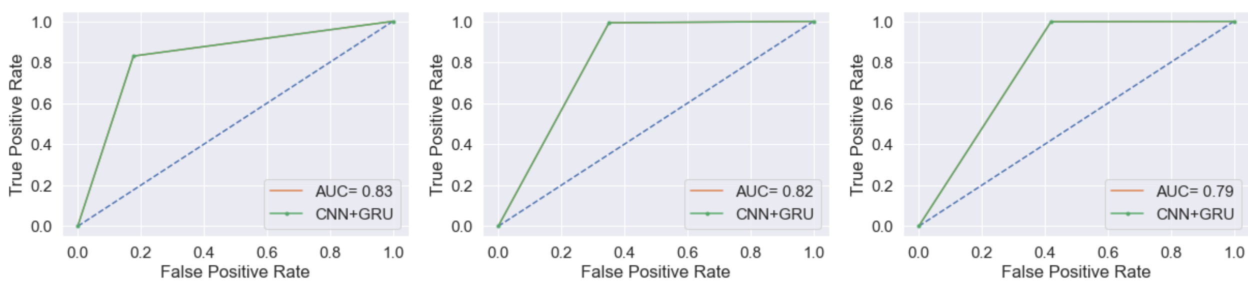
TABLE II: Evaluation Metrics

Dataset	Proposed Model	*Accuracy*	*Precision*	*Recall*	*F1-Score*
BanFakeNews	CNN+GRU (with 2,500 instances)	0.82	0.83	0.83	0.83
	CNN+GRU (with 13,000 instances)	0.95	0.94	0.82	0.87
	CNN+GRU (with 49,977 instances)	0.98	0.94	0.79	0.85
	CNN+LSTM (with 49,977 instances)	0.98	0.89	0.87	0.88
Fake News	CNN+GRU (with 20,800 instances)	0.98	0.99	0.99	0.99



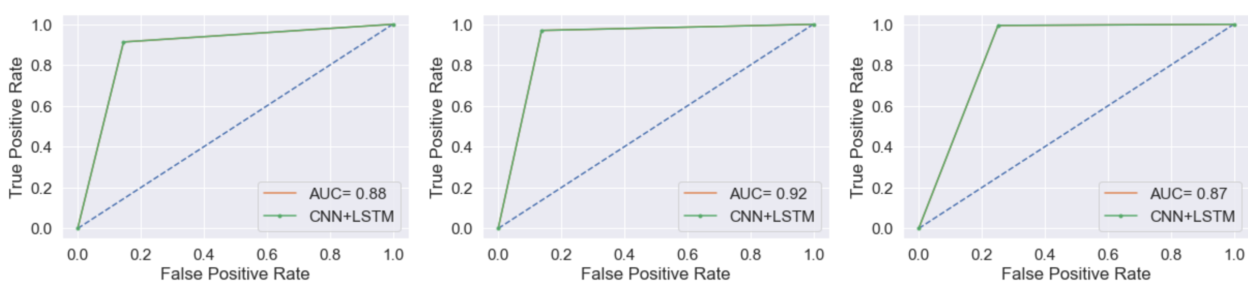
(a) Accuracy on "BanFakeNews" (b) Loss on "BanFakeNews" (c) Accuracy on "Fake News" (d) Loss on "Fake News"

Fig. 4: Accuracy and Loss (During Training and Validation) of CNN+GRU Model on "BanFakeNews" and "Fake News" datasets.



(a) with 2,500 instances (b) with 13,000 instances (c) with 49,977 instances

Fig. 5: This figure illustrates the ROC curve and AUC score of the CNN+GRU model with a different range of instances of the "BanFakeNews" dataset.



(a) with 2,500 instances (b) with 13,000 instances (c) with 49,977 instances

Fig. 6: This figure illustrates the ROC curve and AUC score of the CNN+LSTM model with a different range of instances of the "BanFakeNews" dataset.

V. Experiments and Results

All codes are executed in Python 3.8, using TensorFlow 2.3.0. All experiments are performed on a CoreTM processor Intel CPU i5-7300H 2.50 GHz with 8 GB RAM. The CNN+GRU model's training time is 63 minutes with 10 epochs with the "BanFakeNews' dataset and 29 minutes with the "Fake News" dataset. Adam optimizer is utilized to monitor the learning rate and weights to reduce loss. We have performed several experiments with the combination of CNN and GRU. Our models' average values of performance evaluation metrics are given in table II. In the first experiment, extracted features are fed to CNN+GRU architecture with

2,500 instances of the "BanFakeNews" dataset. In the second experiment, we have trained CNN+GRU architecture with 13,000 instances. Lastly, the proposed ensemble model of CNN+GRU is trained with 49,977 instances. We have also implemented a CNN+LSTM ensemble model for comparison with our proposed model following the same process. While training with a different range of instances, it is observed that the model performed better with 49,977 instances with 98.71% accuracy. The result demonstrates that when the dataset is imbalanced or the false news and true news variance increases, the recall decreases. The loss and accuracy graphs of our best-performed model (CNN+GRU) on both datasets are given in Fig. 4. Finally, CNN+GRU architecture is trained with the balanced "Fake News" dataset and achieved an accuracy of 98.94% which outperformed existing models that used the same dataset. The CNN+GRU model achieved the highest F1-score of 0.99 on the "Fake News" dataset. The ROC and AUC score graphs of our models with different instances are given in Fig. 5 and 6. Besides, the accuracy compared with existing models and our model (CNN+GRU) on the "Fake News" dataset is given in Table III.

TABLE III: This table evaluates the performance comparison with existing models on the "Fake News" dataset.

Dataset	***Model***	***Accuracy***
Fake News	DT+LR+BGC [12]	88.08%
	LSTM+CNN [16]	94.71%
	Merged CNNs [18]	96%
	LSTM [14]	97%
	FNDNet [11]	98.36%
	Proposed Model (CNN+GRU)	98.94%

VI. Conclusion

Fake News Detection is critical for determining whether or not a piece of news is genuine. We explored different neural networks, CNN, GRU, and LSTM, and built an ensemble model using CNN and GRU to detect fake news. We have used pre-trained GloVe embedding as our word embedding because it complements the training process significantly than the traditional bag of words method. It provides each word with a vector projection and the relationship, similarities, differences with other words in the vocabulary. We have trained our model using "BanFakeNews" and "Fake News" datasets. The results illustrate that the proposed method achieved an accuracy of 98.71% using "BanFakeNews" and 98.94% using "Fake News." We have utilized various performance evaluation parameters like F1-score, precision, recall, AUC, ROC, etc., for validating the results. Notwithstanding the excellent performance of our model, there is scope for progression. The Bengali dataset that we have used is imbalanced. We plan to collect and include more Bengali fake news. We will also incorporate lexical features in the future attempt.

References

[1] Arne Roets et al. fake news: Incorrect, but hard to correct. the role of cognitive ability on the impact of false information on social impressions. *Intelligence*, 65:107–110, 2017.

[2] Nir Grinberg, Kenneth Joseph, Lisa Friedland, Briony Swire-Thompson, and David Lazer. Fake news on twitter during the 2016 us presidential election. *Science*, 363(6425):374–378, 2019.

[3] Xinyi Zhou and Reza Zafarani. A survey of fake news: Fundamental theories, detection methods, and opportunities. *ACM Computing Surveys (CSUR)*, 53(5):1–40, 2020.

[4] Supanya Aphiwongsophon and Prabhas Chongstitvatana. Detecting fake news with machine learning method. In *2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pages 528–531. IEEE, 2018.

[5] Muhammad Umer, Zainab Imtiaz, Saleem Ullah, Arif Mehmood, Gyu Sang Choi, and Byung-Won On. Fake news stance detection using deep learning architecture (cnn-lstm). *IEEE Access*, 8:156695–156706, 2020.

[6] M. G. Hussain, M. Rashidul Hasan, M. Rahman, J. Protim, and S. Al Hasan. Detection of bangla fake news using mnb and svm classifier. In *2020 International Conference on Computing, Electronics Communications Engineering (iCCECE)*, 2020.

[7] Marina Danchovsky Ibrishimova and Kin Fun Li. A machine learning approach to fake news detection using knowledge verification and natural language processing. In *International Conference on Intelligent Networking and Collaborative Systems*, pages 223–234. Springer, 2019.

[8] Mykhailo Granik and Volodymyr Mesyura. Fake news detection using naive bayes classifier. In *2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON)*, pages 900–903. IEEE, 2017.

[9] Pritika Bahad, Preeti Saxena, and Raj Kamal. Fake news detection using bi-directional lstm-recurrent neural network. *Procedia Computer Science*, 165:74–82, 2019.

[10] Sherry Girgis, Eslam Amer, and Mahmoud Gadallah. Deep learning algorithms for detecting fake news in online text. In *2018 13th International Conference on Computer Engineering and Systems (ICCES)*, pages 93–97. IEEE, 2018.

[11] Rohit Kumar Kaliyar, Anurag Goswami, Pratik Narang, and Soumendu Sinha. Fndnet–a deep convolutional neural network for fake news detection. *Cognitive Systems Research*, 61:32–44, 2020.

[12] S. Sangamnerkar, R. Srinivasan, M. R. Christhuraj, and R. Sukumaran. An ensemble technique to detect fabricated news article using machine learning and natural language processing techniques. In *2020 International Conference for Emerging Technology (INCET)*, pages 1–7, June 2020.

[13] Saqib Hakak, Mamoun Alazab, Suleman Khan, Thippa Reddy Gadekallu, Praveen Kumar Reddy Maddikunta, and Wazir Zada Khan. An ensemble machine learning approach through effective feature extraction to classify fake news. *Future Generation Computer Systems*, 117:47–58, 2021.

[14] Arush Agarwal and Akhil Dixit. Fake news detection: An ensemble learning approach. In *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, pages 1178–1183. IEEE, 2020.

[15] Muhammad Zubair Asghar, Ammara Habib, Anam Habib, Adil Khan, Rehman Ali, and Asad Khattak. Exploring deep neural networks for rumor detection. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–19, 2019.

[16] Aman Agarwal, Mamta Mittal, Akshat Pathak, and Lalit Mohan Goyal. Fake news detection using a blend of neural networks: an application of deep learning. *SN Computer Science*, 1(3):1–9, 2020.

[17] Md Zobaer Hossain, Md Ashraful Rahman, Md Saiful Islam, and Sudipta Kar. Banfakenews: A dataset for detecting fake news in bangla. *arXiv preprint arXiv:2004.08789*, 2020.

[18] Belhakimi Mohamed Amine, Ahlem Drif, and Silvia Giordano. Merging deep learning model for fake news detection. In *2019 International Conference on Advanced Electrical Engineering (ICAEE)*, pages 1–4. IEEE, 2019.