A Corpus-Based Approach to Bengali Misinformation Detection Using Hybrid Deep Learning Architectures

Meherunnesa Hossain Ibnath1, [a)](#_bookmark0), Saidur Rahman Sujon1, [b)](#_bookmark0), Md. Jakir Hossen2, [c)](#_bookmark0), and M.F. Mridha3, [d)](#_bookmark0)

*1Department of Computer Science and Engineering, Ahsanullah University of Science and Technology, Dhaka, Bangladesh.*

*2Department of Robotics and Automation, Faculty of Engineering and Technology, Multimedia University, Melaka, Malaysia.*

*3Department of Computer Science, American International University Bangladesh, Dhaka, Bangladesh.*

*a) Corresponding author: meherunnesa.ibnath20@gmail.com*

b)sr.sujon.cyb@gmail.com

c)jakir.hossen@mmu.edu.my

d)firoz.mridha@aiub.edu

**Abstract.** The rapid spread of misinformation, disinformation, and fake news has become a significant issue in the modern digital era, with serious consequences for public decision making and trust. This paper proposes a robust framework to identify fake information using a number of datasets that include fake news, spam, misinformation, and rumors. The datasets were intensely preprocessed with text cleaning, augmentation, and feature extraction using Term Frequency-Inverse Document Frequency (TF- IDF). Different ML and DL algorithms were tried, including KNN, SVM, NB, DT, LR, Bi-LSTM, RBM, RNN, DBN, CNN-LSTM, CNN-GRU, and transformer models (mBERT, BanglaBERT, Bangla-Electra). DBN resulted in the highest accuracy of 89. 92%, followed by CNN-GRU with an accuracy of 89.43% and CNN-LSTM at 89.30%, then by Decision Tree with an accuracy of 88.83%. The transformer models produced variable outputs, BanglaBERT (85. 16%), mBERT (75. 00%) and Bangla-Electra (53. 36%). The study enlists the promise of newer DL approaches toward efficient real-time scalable detection of disinformation and presents a foundation for future research in domain-specific data, contextual embeddings, and efficient designs towards high accuracy and deployment effectiveness.

## INTRODUCTION

Disinformation and fake news are now pandemics that have wreaked havoc on the public trust and threatened democratic institutions. With social networks now becoming more of an information source, detection and countermeasures against disinformation are now more critical than ever before. In the task of identifying misinformation, there have been highly sophisticated deep learning models like BERT, BiLSTM, and CNN with good performance results [[1](#_bookmark7)], but issues of class imbalance and lack of language-specific datasets persist [1a]. English is the focus of most of the studies, with Bengali and other languages being under-researched. This paper addresses such research gaps using CNN-GRU in order to improve detection accuracy and class balance distributions efficiently. Misinformation detection, particularly for Bengali, involves numerous challenges that this study aims to address. While architectures like LSTM and CNN have shown promise, hybrid models like CNN-GRU are not yet fully explored for misinformation detection [[1](#_bookmark7)], [[3](#_bookmark9)]. Class imbalance in data sets has a tendency to produce biased predictions. This paper solves this issue by using random oversampling techniques to improve the generalization ability of models [[4](#_bookmark12)], [[5](#_bookmark13)]. Bengali, despite being one of the most spoken languages globally, does not have sufficient research on detecting misinformation. This paper enhances Bengali misinformation detection using hybrid, deep learning, and traditional models, addressing class imbalance and feature selection.

## LITERATURE REVIEW

Misinformation, disinformation, and fake news detection have been a subject of extensive interest in the past few years, and many studies have explored various methods. This section presents a review of significant contributions to the field, such as developments and challenges of misinformation detection.

Reshi and Ali [[1](#_bookmark7)] benchmarked models such as BERT, BiLSTM, BiGRU, and CNN for Reddit misinformation detection with the best accuracy of 90.62% using MPNet-SNLI embeddings and a DNN. This demonstrates the power of transfer learning and good quality contextual embeddings for classification. Rashid et al. [[4](#_bookmark8)] demonstrated the effectiveness of LSTM and Bangla BERT for Bangla misinformation detection with 98.77% accuracy. Toma and Huleihel [[3](#_bookmark9)] proposed two new algorithms for multiclass classification using Multiple Sequential Probability Ratio Test (MSPRT) and Graph Neural Networks (GNNs) that achieved superior performance over state-of-the-art meth- ods. Puri [[5](#_bookmark10)] reported a detailed comparison of fake news detection methods by comparing traditional machine learning methods like Logistic Regression and SVM with state-of-the-art LLM-based methods. They refer to the promise and challenges of combining technological innovation and media literacy to counter misinformation. Islam et al. [[6](#_bookmark11)] surveyed DL techniques for misinformation detection, asserting their superiority to traditional meth- ods in handling complex, imbalanced data. The authors further discussed possible future research directions, such as incorporating user mental health into detection models. Zhu et al. [[7](#_bookmark12)] proposed the MSHR-FCSSVM hybrid method- ology for imbalanced data classification using Mahalanobis distance-based resampling and cost-sensitive SVM. Their methodology gained significant performance on 20 imbalanced datasets. Cabral et al. [[8](#_bookmark13)] tackled Brazilian Por- tuguese WhatsApp message misinformation detection using TF-IDF with bigrams and trigrams with an F1 score of

0.87. Mugdha et al. [[9](#_bookmark14)] explored fake news detection in Bengali using Gaussian Naive Bayes, achieving 87% accu- racy through feature extraction with TF-IDF and Extra Tree Classifier. Sharma et al. [[10](#_bookmark15)] compared XGBoost and LSTM for fake news detection, finding that LSTM outperformed XGBoost, achieving 99.9% accuracy. Together they underscore the growing need to devise machine learning and deep learning methods to combat the challenges of misinformation detection both within and across languages and platforms.

## DATASET

This section describes the datasets used to detect misinformation, disinformation, fake news, spam, and rumors. In order to offer a balanced evaluation, we prepared an integrated corpus by aggregating multiple datasets. The merged dataset enhances quality and diversity so that robust classification is possible in the desired categories.

1. *Misinformation, Disinformation, and Fake News Detection Dataset:*This dataset contained a total of 12,903 records with the null values eliminated before processing. For better text readability, extractive summariza- tion was utilized by selecting prominent sentences to form a concise summary (see Algorithm [1](#_bookmark1)). Dataset link:[Misinformation Dataset](https://www.kaggle.com/datasets/hrithikmajumdar/misinformation-disinformation-fake-news-dataset). It is composed of Bengali texts labeled as Misinformation (2,182), Disinformation (3,017), and Fake News (6,670). The distribution of the labels is presented in Figure [1](#_bookmark2).

**Algorithm 1** Extractive Summarization Algorithm

**Require:** Text, *num*\_*sentences*, *word*\_*limit* **Ensure:** Extractive summary of the input text

1: Tokenize the text into sentences.

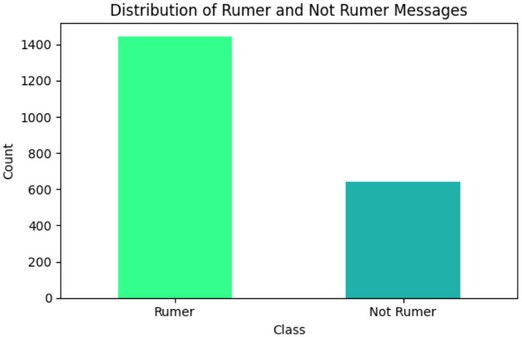
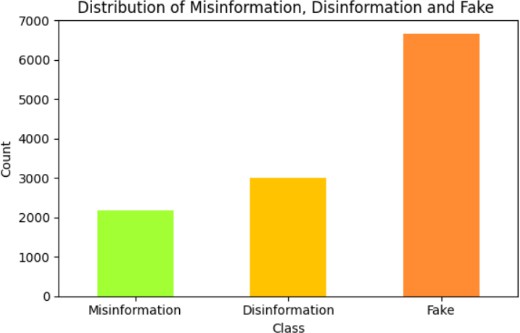
2: Compute TF-IDF vectors for all sentences.

3: Calculate cosine similarity between sentence pairs.

4: Score sentences by summing similarity scores.

5: Select top *num*\_*sentences* based on scores. 6: Combine selected sentences into a summary. 7: Truncate the summary to *word*\_*limit* words. 8: **Return** the truncated summary.

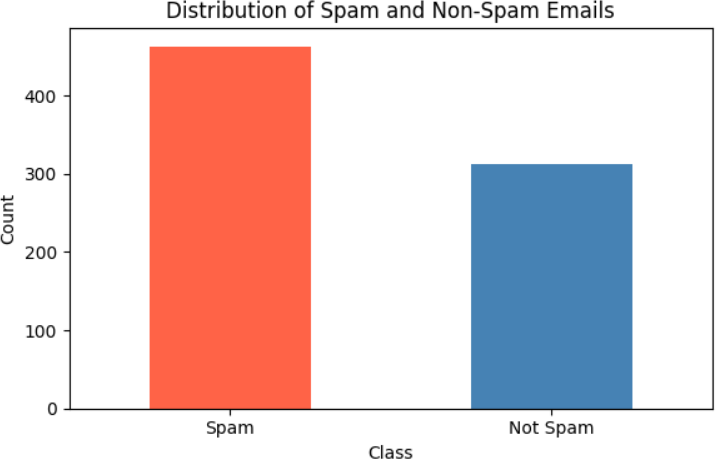
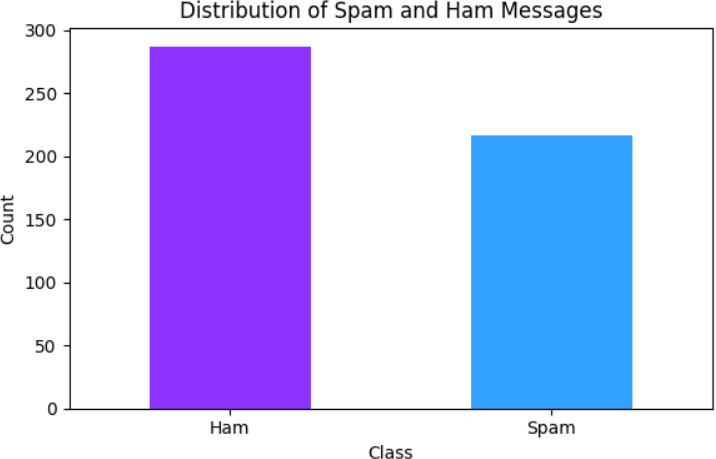
1. *Rumor Detection Dataset:*This data set contains 2,088 Bengali labeled text samples. Null values were re- moved to ensure the data was complete. Dataset link: Abu Naim Dataset. The label distribution is Rumor 1,445 entries, Not Rumor 641 entries. The label distribution is shown in Figure [1](#_bookmark2).
2. *Bangla Spam Detection Dataset:*The dataset consists of spam messages label distribution is shown in Figure [2](#_bookmark3) and merges two sources:
   * **Dataset 1:** [Bangla Spam SMS Dataset](https://www.kaggle.com/datasets/faribatasniakhan/bangla-spam-sms). This dataset contains 504 records. Following the removal of null values Ham 287 records, Spam 217 records.
   * **Dataset 2:** [Bangla Spam Email Dataset](https://www.kaggle.com/datasets/durjoymistry/bangla-spam-email). It began at 5,730 entries, and we’ve employed Extractive Sum- marization here too as our text were too lengthy. On removing null values, we had 776 entries remaining with the label count as Spam 463 entries, Not Spam 313 entries. Combining both datasets resulted in 680 spam entries for the unified corpus.



(a) Misinformation, Disinformation, (b) Rumor and Not Rumor Labels

and Fake News Labels

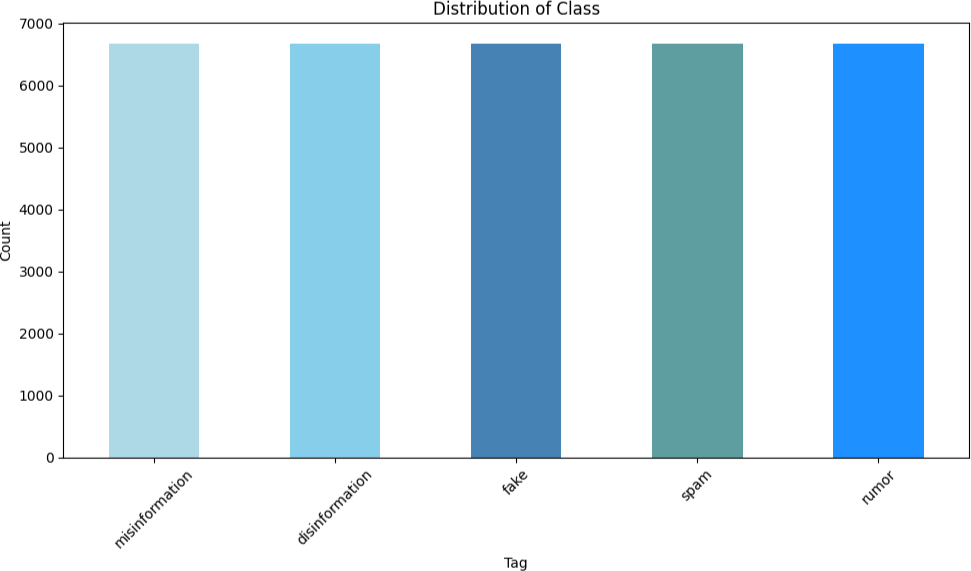
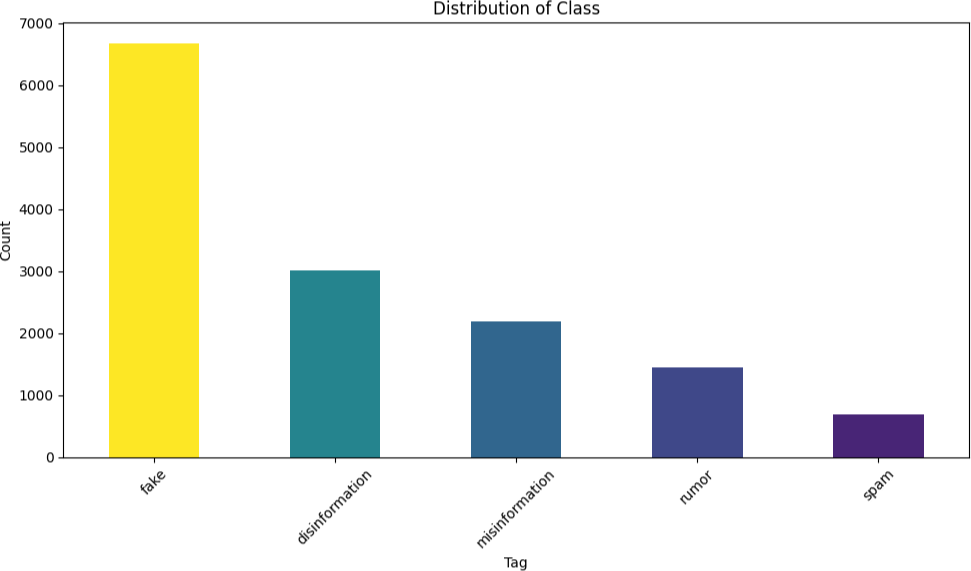
**FIGURE 1.** Distribution of misinformation and rumor labels



(a) Bangla Spam SMS Dataset (b) Bangla Spam Email Dataset

**FIGURE 2.** Distribution of Bangla spam SMS and email datasets

The data sets were merged into a homogeneous five-tag corpus: Misinformation (Label 0), Disinformation (Label 1), Fake News (Label 2), Spam (Label 3), and Rumors (Label 4). The previous dataset contained 13,994 samples out of which 6,670 samples were of Fake News label, 3,017 samples of Disinformation label, 2,182 samples of Misinformation label, 1,445 samples of Rumor label, and 680 samples of Spam label. Homogeneous data set was highly class imbalanced. For this, we employed random oversampling, replicating instances of the minority classes to balance every label’s representation. Oversampling mitigates imbalance but may increase the risk of overfitting. Nevertheless, the approach improves classification of under-sampled classes. After oversampling, dataset size was raised to 33,350 samples and 6,670 samples per label (see Figure 3).



(a) Class Distribution in the Unified Corpus (b) Dataset Distribution After Oversampling

**FIGURE 3.** Comparison of class distribution before and after oversampling

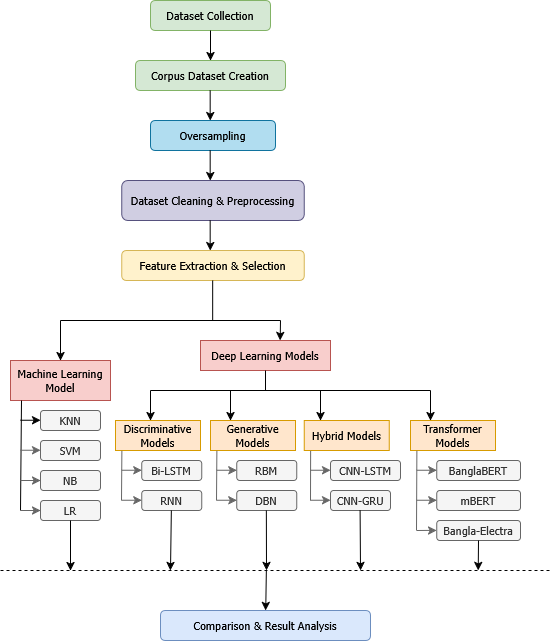
Table [1](#_bookmark4) summarizes all the datasets we collected and utilized for our study, including their sources, record counts after cleaning, and class labels.

**TABLE 1.** Summary of datasets used

|  |  |  |  |
| --- | --- | --- | --- |
| **Dataset Name** | **Source** | **Number of Records** | **Classes / Labels** |
| Misinformation, Disinforma-  tion, Fake News Dataset | Kaggle | 12,903 (after cleaning) | Misinformation, Disinformation,  Fake News |
| Rumor Detection Dataset | Kaggle | 2,088 (after cleaning) | Rumor, Not Rumor |
| Bangla Spam SMS Dataset | Kaggle | 504 (after cleaning) | Ham, Spam |
| Bangla Spam Email Dataset | Kaggle | 776 (after cleaning) | Spam, Not Spam |
| **Unified Corpus (Merged)** | Aggregated | 13,994 (before  oversampling) | Misinformation, Disinformation,  Fake News, Rumor, Spam |
| **Unified Corpus (After Over-**  **sampling)** | Aggregated | 33,350 | Misinformation, Disinformation,  Fake News, Rumor, Spam |

## RESEARCH METHODOLOGY

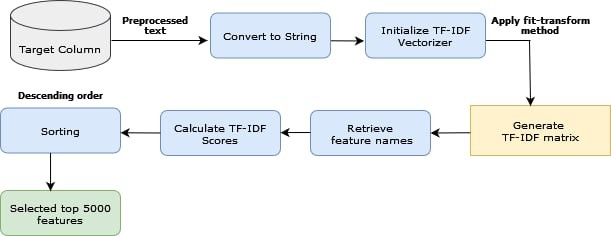
The research methodology is a step-by-step one, describing every phase involved in the construction and evaluation of the models. The general methodology is presented below, from data collection through implementation and testing of the models. The Figure [4](#_bookmark5) illustrates our complete methodology.

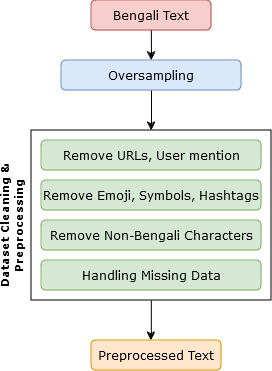


**FIGURE 4.** Overview of methodology

## Data Cleaning and Preprocessing and Feature Extraction & Selection

The collected dataset was subjected to a series of preprocessing steps to enhance its quality and check its suitability for further analysis. To fight the class imbalance issue, oversampling techniques were employed, as discussed in the Oversampling section. Then, a series of preprocessing tasks were carried out to further clean the dataset. Initially, URLs, user mentions, emojis, and hashtags were removed as these do not convey any useful information for misinformation detection. Next, special characters and non-Bengali characters were also removed to maintain the integrity of the language-specific information. Lastly, missing values were handled by either removing incomplete records or inputting missing values where necessary, making the dataset uniform.

In feature extraction, we used the Term Frequency-Inverse Document Frequency (TF-IDF) method, which sorts words based on their frequency in a document versus their frequency in the entire dataset. For our task, we selected features by their highest TF-IDF scores, and we retained only words that are most informative and descriptive in context. This selective approach enhances model performance because it only emphasizes the most prominent terms with a high weight in distinguishing between different classes of misinformation (see Figure 5).



(1) Preprocessing Technique (2) TF-IDF Feature Extraction Technique

**FIGURE 5.** Preprocessing and feature extraction techniques

## MODEL IMPLEMENTATION

In the present work, we employed a wide range of machine learning and deep learning models for the classification task. Models are categorized based on their paradigms and structural architectures. Below is an exhaustive description of the models that were implemented:

* **K-Nearest Neighbors (KNN):** Distance-based classifier that makes predictions of class labels from the vote of the *k* nearest neighbors in feature space.
* **Support Vector Machine (SVM):** Finds a best hyperplane with the goal of maximizing margin between classes in high-dimensional space.
* **Decision Tree (DT):** A tree-based model that recursively splits the data based on feature values to minimize entropy or Gini impurity.
* **Multinomial Naïve Bayes (MNB):** A feature-independent probabilistic classifier, which is typically employed for text categorization.
* **Logistic Regression (LR):** A linear model that outputs the probability of a point belonging to a specific class by a logistic function.
* **Restricted Boltzmann Machine (RBM):** A neural network that learns joint probability distribution between hidden and visible layers for unsupervised feature learning.
* **Deep Belief Network (DBN):** A hierarchical model composed of stacked RBMs that captures higher-level abstractions in data.
* **Recurrent Neural Network (RNN):** Designed for sequence modeling by capturing temporal dependencies in sequential data.
* **Bidirectional LSTM (BiLSTM):** One of the variants of LSTM that takes data in both the forward as well as backward directions to enhance contextual understanding.
* **CNN-LSTM:** Uses Convolutional Neural Networks (CNN) to extract features and LSTM to detect temporal relationships.
* **CNN-GRU:** Combines CNN for spatial feature learning with Gated Recurrent Unit (GRU) to learn sequential patterns, ideally suited for sequence classification task.
* **BanglaBERT:** BERT-based transformer model pre-trained specifically in Bengali text to facilitate improved performance on Bengali NLP tasks.
* **Bangla-Electra:** A transformer-based model trained with the Electra method, further fine-tuned for improved language modeling capabilities in Bengali.

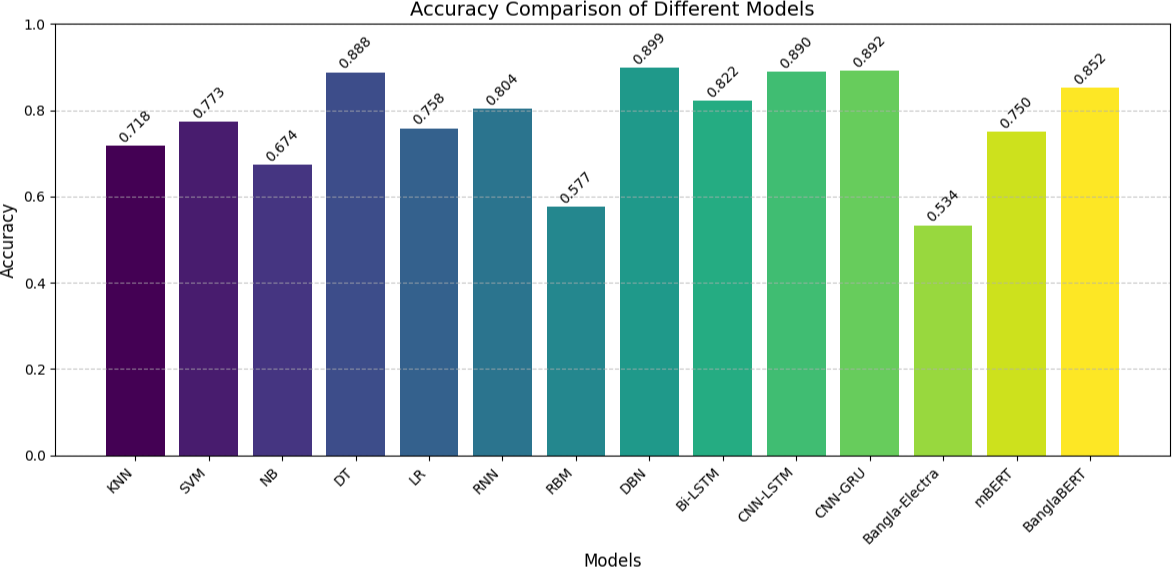
All these models were selected based on their applicability for text classification and ability to handle linguistic and contextual complexities within the dataset.

## Evaluation Metrics

To comprehensively evaluate the performance of the classification models, we employed a number of the most widely used evaluation measures: Accuracy, Precision, Recall, F1 Score, and Error Rate. Accuracy is employed to quantify the ratio of overall correct predictions, true positives and true negatives, to all of the predictions, indicating the general accuracy of the model. Accuracy is the proportion of correctly predicted positive instances to all positive instances predicted, which indicates the ability of the model in avoiding false positives. Recall or sensitivity or the true positive rate indicates the ability of the model in correctly identifying the actual positive instances, thus its ability in avoiding false negatives. The F1 Score, the harmonic mean of precision and recall, presents a balanced measure particularly helpful when handling imbalanced sets. Last but not least, the Error Rate indicates the proportion of incorrect predictions to total predictions, and is the complementary measure of accuracy. The measures together provide a general estimate of the model’s performance in detecting misinformation.

# RESULTS ANALYSIS AND DISCUSSION

This section presents an extensive comparison of the performance of eight learning models with the TF-IDF feature extraction technique. We evaluate and compare models in terms of various measures like accuracy, precision, recall, F1-score, and error rate. Figure [6](#_bookmark6) shows the results of our research.



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision** | **Recall** | **F1-Score** | **Error Rate** |
| KNN | 0.7021 | 0.7462 | 0.7021 | 0.6838 | 0.2979 |
| SVM | 0.7748 | 0.7719 | 0.7748 | 0.7719 | 0.2252 |
| Naive Bayes | 0.6870 | 0.7030 | 0.6870 | 0.6884 | 0.3130 |
| Decision Tree | 0.8883 | 0.8887 | 0.8883 | 0.8858 | 0.1117 |
| Logistic Reg. | 0.7583 | 0.7521 | 0.7583 | 0.7538 | 0.2417 |
| Bi-LSTM | 0.8250 | 0.8234 | 0.8250 | 0.8229 | 0.1750 |
| RBM | 0.5773 | 0.5620 | 0.5773 | 0.5661 | 0.4227 |
| RNN | 0.8037 | 0.8004 | 0.8037 | 0.7994 | 0.1962 |
| **DBN** | **0.8992** | **0.8998** | **0.8992** | **0.8973** | **0.1007** |
| CNN-LSTM | 0.8930 | 0.8918 | 0.8930 | 0.8913 | 0.1070 |
| CNN-GRU | 0.8943 | 0.8939 | 0.8943 | 0.8923 | 0.1057 |
| Bangla-Electra | 0.5336 | 0.4429 | 0.5336 | 0.4138 | 0.4664 |
| mBERT | 0.7500 | 0.7519 | 0.7500 | 0.7490 | 0.2500 |
| BanglaBERT | 0.8516 | 0.8596 | 0.8516 | 0.8500 | 0.1484 |

(1) Performance Metrics for Models (2) Accuracy Comparison of Models

**FIGURE 6.** Performance analysis of classification models

The Deep Belief Network (DBN) was the best at 89.92%, followed by CNN-GRU (89.43%) and CNN-LSTM

(89.30%). Deep learning models performed better than traditional models such as KNN (70.21%), Naive Bayes (68.70%), and SVM (77.48%) that failed to catch subtle data patterns. Deep Belief Network (DBN) confusion matrix shows good classification with high true positive values along the diagonal. There are however some misclassifications, mostly in class 2, showing scope for improvement in subtle case handling.

Both CNN-GRU and CNN-LSTM worked well. These hybrid models utilize CNN for extracting local features and RNNs (GRU/LSTM) to model temporal dependency. CNN-GRU performs well with imbalanced data, whereas CNN-LSTM performs best with sequential long-term data and hence excels in some cases. Transformer models perform better than the traditional RNN-based models in sentiment analysis. mBERT achieves a score of 75.00%, outperforming Logistic Regression (75.83%) but falling behind Bi-LSTM (82.50%). BanglaBERT reaches 85.16%, performing better than CNN-GRU (89.43%) and CNN-LSTM (89.30%). Bangla-Electra reaches 53.36%, which is lower than the performance of other models. The performance confirms the superiority of Bengali-specialized transformers over multilingual models.

SVM and Decision Tree performed well with accuracies of 77.48% and 88.83%, respectively. Decision Trees are efficient in representing complex decision boundaries but tend to overfit. SVM, while being good, could not keep pace with the performance of the deep learning models in representing complex relationships. Naive Bayes and KNN were the worst performers with accuracies of 68.70% and 70.21%. Naive Bayes did not perform well with TF-IDF features because of its independence assumption of the features, while KNN took a poor time in high-dimensional spaces and provided lower accuracy. With respect to error rate, DBN had the minimum error rate of 0.1007, followed by CNN-LSTM (0.1070) and then CNN-GRU (0.1057), which showed improved classification accuracy for real-world applications.

# CONCLUSION AND FUTURE WORK

In this study, various machine learning and deep learning models used for identifying misinformation have been compared based on TF-IDF feature extraction. Among them, hybrid deep learning models with Deep Belief Network (DBN) provided the maximum outcome for identifying complex textural patterns at 89.92% rate. Whereas traditional machine learning models handled low-level data sets successfully, they were not easily able to deal with complexity found in linguistic along with contextual structure of disinformation. High-quality domain-specific data collection is among future works. Research of large language models like Deepseek and GPT can enhance semantic understanding, whereas models like T5, XLNet, and GPT-4 offer promising directions. Additionally, pre-processing technique optimization and model fine-tuning to meet real-time requirements will increase scalability in applications like social media tracking and news authentication.

# REFERENCES

1. J.A. Reshi and R. Ali, “Leveraging transfer learning for detecting misinformation on social media,” International Journal of Information Technology **16** (2), 949–955 (2024).
2. A. Khan, K. Khan, W. Khan, S.N. Khan, and R. Haq, “Knowledge-based Word Tokenization System for Urdu,” Journal of Informatics and Web Engineering 3(2), 86–97 (2024).
3. D. Toma and W. Huleihel, “Sequential Classification of Misinformation,” (2024). https://doi.org/10.48550/arXiv.2409.04860
4. M.R.A. Rashid, R. Roy, D.M.S. Rahman, M.A. Saleh, A.A.H. Khan, Md. Abu Rayhan, K.F. Ahmed, N. Monsoor, and M. Hasan, “A Comprehensive Dataset and Deep Learning Approach for Misinformation Detection on Social Media in Bangladesh,” International Journal of Computer and Data Science **18** (1), 1–10 (2025).
5. Ridham Puri, “Fake News Detection: A Comprehensive Study on Analysis and Modern Classification Techniques,” (2024). http://dx.doi.org/10.13140/RG.2.2.23779.16164
6. M.R. Islam, S. Liu, X. Wang, and G. Xu, “Deep learning for misinformation detection on online social networks: a survey and new perspectives,” Social Network Analysis and Mining **10** (1), 82 (2020).
7. B. Zhu, X. Jing, L. Qiu, and R. Li, “An Imbalanced Data Classification Method Based on Hybrid Resampling and Fine Cost Sensitive Support Vector Machine,” Computers, Materials & Continua **79** (3), 3977–3999 (2024).
8. L. Cabral, J. Monteiro, J. Franco Da Silva, C. Mattos, and P. Mourão, “FakeWhastApp.BR: NLP and Machine Learning Techniques for Misinformation Detection in Brazilian Portuguese WhatsApp Messages,” in *Proceedings of the 23rd International Conference on Enterprise Information Systems* (SCITEPRESS – Science and Technology Publications, Online Streaming, 2021), pp. 63–74.
9. S.B.S. Mugdha, S.M. Ferdous, and A. Fahmin, “Evaluating Machine Learning Algorithms for Bengali Fake News Detection,” in *2020 23rd International Conference on Computer and Information Technology (ICCIT)* (IEEE, Dhaka, Bangladesh, 2020), pp. 1–6.
10. S. Sharma, M. Saraswat, and A.K. Dubey, “Fake News Detection Using Deep Learning,” in *Knowledge Graphs and Semantic Web*, edited by B. Villazón-Terrazas, F. Ortiz-Rodríguez, S. Tiwari, A. Goyal, and M. Jabbar (Springer International Publishing, Cham, 2021), pp. 249–259.