Journal of Applied Technology and Innovation
Vol. 10, Issue 1, 2026, pp. 40-61
e-ISSN:2600-7304-ISSN
doi: https://doi.org/10.65136/jati.v10i1.325

Explainable AI-Driven Bangla News Classification: Comparative Study of ML and DL Approaches

Jahangir Hussen[1], Mohammad Rashed Hasan Polas [2]*, MST. Najnin Akter Emu[1], MD Yusuf Mia[1], Md Sifat Mahmud[1], Md Mazharul Haque Emon[1], Rakib Chowdhury[1]

[1] Department of Computer Science and Engineering, Sonargaon University, 147/I, Green Road Panthapath, 1215, Dhaka, Bangladesh
[2] Department of Business Administration, Sonargaon University, 147/I, Green Road Panthapath, 1215, Dhaka, Bangladesh

*Corresponding author: rashedhasanpalash@gmail.com

Received: 2026-02-25; Accepted: 2026-03-19; Published: 2026-04-08

Abstract

Classifying Bangla news is still a big problem in processing languages with few resources, however most recent studies emphasize performance metrics instead of model explainability. This study introduces an explainable AI-based system for the multiclass classification of Bangla news articles into four categories: Sports, International, Entertainment, and National, utilizing a dataset of 11,904 articles. We conducted a comprehensive comparison of classical machine learning (ML) and deep learning (DL) methodologies. We used Ridge Classifier, SGD Classifier, Multinomial Naive Bayes, and XGBoost to help us learn how to use machines. We got the best score of 92.55% accuracy by using a hard-voting group of the four best linear models. We make deep learning models like CNN-LSTM, Simple LSTM, Stacked LSTM, and GRU. When they work together, the three best models get 95.76% right, and when they work alone, they get 95.46% right. We used LIME (Local Interpretable Model-agnostic Explanations) to make the Ridge Classifier and Multinomial Naive Bayes models more accurate and easier to understand. The results show that DL ensembles are better at finding the right answer than more common ML models. However, linear ML models are faster and easier to learn, train, and use. This change made Bangla NLP systems more open and reliable. This is a great way to stay up to date on the news and learn about what's going on in places where Bengali is spoken.

Keywords: *Bangla news classification, Explainable AI, Machine Learning, Deep Learning, Ensemble techniques, Low-resource language*

1. Introduction

Digital news ecosystems are quickly growing in languages like Bangla that lack resources. We need better, more accurate, and easier ways to sort through a lot of unstructured text. A lot of people speak Bangla, but computers do not understand it. More than 300 million people speak it as their first language. Most of them live in Bangladesh, West Bengal, and Tripura in India. Bangladeshis also live in a lot of other

places, like the Middle East, North America, Europe, and Southeast Asia. For a lot of practical reasons, it is important to automatically sort Bangla news into clear categories like Sports, International, Entertainment, and National. Some of these are systems that gather and filter news in real time, suggest content based on what you like, search engines that focus on certain topics, automated pipelines for finding false information and misinformation, sentiment-aware media monitoring tools, and smart content moderation systems used on social media sites, news sites, and digital archives for Bengali speakers (Rahman et al., 2021).

Natural language processing (NLP) has come a long way for languages like English, Mandarin, and Spanish, which have a lot of information available. Bangla NLP cannot move progress because its structure and resources are not very good. The grammar of the language has a lot of inflection and agglutination. It has hard conjunct characters, a flexible word order, and a lot of case markers and honorifics. All of these things make it hard to do stemming, lemmatization, tokenization, and syntactic parsing. There aren't many big datasets with good annotations for languages other than Hindi, Tamil, or English, which are all Indic languages. There are many Bangla corpora that anyone can use, but most of them are small, only cover one area, make a lot of noise, or imbalanced. BanglaBERT and SahajBERT are two language models that can already read and write in Bangla. There are fewer as Bangla models than English models, and the Bangla models are usually smaller.

Most of the current research on how to organize Bangla news stories is centered on two main ideas. The first one uses standard machine learning pipelines to get sparse, hand-crafted features (mostly TF-IDF or word frequency vectors) and put them into well-known algorithms like Support Vector Machines, Naive Bayes variants, Logistic Regression, Random Forests, and gradient boosting methods like XGBoost. These strategies are easy to understand and usually work well as a first step, especially if the data is clean and balanced. For the second method, deep learning methods like convolutional neural networks (CNN) and recurrent architectures (LSTM, GRU, and BiLSTM) are used. Most of the time, these methods use simple word embeddings that were either learned from scratch or taken from a few Bangla fastText sources. Deep models are better at getting things right, especially when it comes to long articles that need some background. But they still tend to overfit on small datasets, are costly to run, and, most importantly, are difficult to understand (Ahmad et al., 2022).

There is a growing need for explainable artificial intelligence (XAI) as more and more AI-powered tools are used to make decisions in media and information systems that are open to the public. People are worried about prejudice spreading, accountability, and discrimination that should not happen when "black-box" models aren't clear in important areas like news classification. Local Interpretable Model-agnostic Explanations (LIME) and other methods for post-hoc interpretability fix these problems by making accurate and easy-to-read surrogate models that show how some complicated classifiers work. LIME finds and ranks the words or phrases that have the most effect on a specific prediction in text classification. This helps reporters, content moderators, researchers, and end users understand why an article was put in that group (Salehin et al., 2021).

This study makes the following principal contributions to the advancement of Bangla NLP:

- A thorough comparison of traditional machine learning and modern deep learning methods for sorting Bangla news stories, using a set of 11,904 articles from Bangladesh's best news sites.

- The development and evaluation of six meticulously selected machine learning classifiers (Logistic Regression, Linear Support Vector Machine (LinearSVC), Ridge Classifier, SGD Classifier, Multinomial Naive Bayes, and XGBoost) and four deep neural architectures (Simple LSTM, Stacked LSTM, GRU, and hybrid CNN-LSTM) utilizing ensemble techniques in both domains to enhance predictive accuracy.

- The methodical use of LIME-based post-hoc interpretability on conventional ML models (Multinomial Naive Bayes and Ridge Classifier) to produce visual explanations and numerical feature importance scores for each instance, clarifying the reasoning behind the decisions rendered.

- A close look at the pros and cons of different performance metrics, such as how well they sort things, how easy they are to understand, how long it takes to train and test them, and how much computing power they need. This will lead to practical suggestions for putting Bangla news classification systems into place in places where resources are limited.

The manuscript structure is as follows. Firstly, previous Bangla text sorting methods, ensemble learning methods, and new advances in explainable AI for processing natural language are discussed. Next, the dataset, the text preprocessing pipeline, feature selection method, list of models used, ensemble methods, experimental protocol, and evaluation metrics were explained. After that results of the quantitative classification, confusion matrix analysis, learning curves, runtime comparisons, and full qualitative interpretations based on the LIME explanations are displayed. Next, key results, implications in media and information systems, study limitations, and future research directions are discussed, followed by the conclusion and summary (Limon et al., 2018).

The main goal of this study is to make sure that AI tools can correctly process and organize Bangla news that is clear, reliable, and fits with the culture. Deep learning ensembles are rarely used in contexts with strict rules because it is hard to explain them, raising moral issues. The proposed framework aims to substantially improve the creation of more dependable, accountable, and user-friendly natural language processing systems specifically designed for the Bangla language context by combining high classification accuracy with stringent post-hoc interpretability (Hossain et al., 2025).

2. Literature Review

Over the last ten years, Bangla natural language processing (NLP) has shown continuous progress particularly in text classification and news categorization. Bangla NLP still lags behind English Mandarin and Spanish because of three main reasons which include the limited amount of data and various types of data that exist and the development of better algorithms and better architectural designs and the current focus on making models open for inspection and providing explanations and building user trust.

Researchers in Bangla NLP during the period from 2015 to 2020 relied on traditional machine learning pipelines to conduct their research. Researchers usually gathered small amounts of data by collecting information from Bangla blogs and online forums and social media posts and a few news websites. The first datasets from this period contained between 200 and 2000 labeled samples at their highest. The feature extraction process used traditional bag-of-words (BoW) models and term frequency-inverse document frequency (TF-IDF) weighting and sometimes n-gram representations. To classify data researchers used Multinomial Naive Bayes and Logistic Regression as their main classification methods. Researchers evaluated Support Vector Machines (SVM) performance by testing its linear kernel and Support Vector Machines (SVM) performance through its linear kernel tests.

The basic studies demonstrated that binary classification tasks (positive versus negative sentiment and fake versus real news) and basic multiclass tasks (sports and politics and entertainment) could achieve satisfactory classification results through the use of basic annotation and low-cost computation methods. The advantages of these methods included quick training and prediction speeds and straightforward implementation and low system requirements and direct feature importance analysis which allowed users to interpret results. The characteristics of these devices proved to be practical for the early phases of Bangla NLP development when organizations lacked access to large labeled datasets and graphical processing unit (GPU) resources (Alam et al., 2021; Sen et al., 2022).

Research scientists began testing advanced traditional machine learning methods after publicly available datasets began to grow in size through community contributions and shared repositories and increasing interest in low-resource NLP. Researchers commonly selected kernel-based SVM models with RBF kernels and Decision Trees and Random Forests and XGBoost and LightGBM and CatBoost as their primary tree-based ensemble methods. The evaluation process for the models involved testing them on multiclass news classification benchmarks which featured thousands to tens of thousands of documents that researchers had collected from major Bangladeshi and West Bengal media outlets.

The period showed that well-tuned tree-based ensemble models and linear models which used regularization techniques could achieve weighted F1-scores between 0.85 and 0.93 on balanced data sets. The methods proved successful for handling Bangla text because they used regularization techniques which kept learning within high-dimensional spaces and noise handling capabilities which could process actual text with code-mixing and dialectal variations and their design which matched with TF-IDF and count-based features. The process of feature selection followed by hyperparameter optimization brought enhanced model performance through better generalization ability (Sen et al., 2021; Chakraborty et al., 2025).

The introduction of deep learning methods created an essential transformation in Bangla text classification research. Studies from 2018 to 2020 used recurrent neural networks which included Long Short-Term Memory (LSTM) units and bidirectional LSTM (BiLSTM) models in their Bangla research. The system designs could establish sequence connections while maintaining sentence and document distance links which traditional bag-of-words systems were unable to accomplish. BiLSTM models performed better than traditional methods when tested on 10,000 to 50,000 sample datasets because they increased F1-scores by 5 to 12 percentage points (Banik et al., 2019).

The text adaptation of Convolutional Neural Networks (CNNs) which included Kim's 2014 architecture became widely used because of its ability to identify text patterns and n-grams and hierarchical features without needing to process sequences. The design of hybrid models combined convolutional layers which extracted features with recurrent layers which handled sequence modeling. The dual design of CNN-LSTM and CNN-BiLSTM created hybrid systems which produced optimal results for both pure Bangla data and data that had undergone partial quality checks with F1-score performance reaching the 90s range for all multiclass evaluations. Deep learning techniques enabled models to learn from raw tokenized text by eliminating the need for manual feature crafting (Pramanik & Noor, 2025).

Researchers used ensemble learning to gain more performance benefits while making their systems more resilient. The classical ensemble method worked by combining outputs from multiple standard classifiers through majority voting or weighted soft voting or stacking meta-learners. The method created steady improvements which produced results that extended between 1 and 4 percent improvements in F1. Researchers built neural architecture ensembles through LSTM and GRU and CNN and Transformer variants which they used to develop new AI-based systems that learned to combine their outputs through two methods: averaging probabilities or learned learning methods. The method of ensemble learning brought the advantage of obtaining helpful inductive biases which showed different behaviors across multiple model types (Mandal & Sen, 2014; Emon et al., 2019).

The Bangla NLP research field has developed better predictive models but researchers have not studied how to make their models understandable to users. The overwhelming majority of published works focus exclusively on standard performance metrics—accuracy, precision, recall, F1-score (macro/micro/weighted), confusion matrices, and occasionally training/inference runtime—while providing almost no insight into why models assign particular labels to specific documents (Roy et al., 2023). The system operates without any public access which creates security dangers because journalists must identify news stories according to established principles that protect confidential information.

The XAI community has adopted post-hoc interpretability methods that include LIME (Local Interpretable Model-agnostic Explanations), SHAP (SHapley Additive exPlanations), Integrated Gradients, and attention visualization as essential tools for understanding how complex models operate. The methods work well for text classification because they show which tokens, phrases, sentences, or contextual patterns most affect the predictions of particular text elements (Ahmad et al., 2022). The initial XAI applications which researchers tested in different low-resource Indic languages (Hindi, Tamil, Bengali variants) discovered that systems failed because they used named entities and emotional trigger words and domain-specific jargon and superficial lexical cues instead of proper semantic understanding.

The field of Bangla research has investigated interpretability methods through only a few studies which applied these techniques to restrict their results to specific categories like binary sentiment classification and fake news detection. Users find post-hoc explanation techniques to provide better insights than attention mechanisms used in Transformer-based models which show attention noise through their weight patterns (Hossain et al., 2020).

The combination of ensemble methods with explainability functions as a research area which remains underexplored in Bangla NLP studies. Ensemble methods work to improve accuracy while decreasing variance but they create challenges for interpretation because their final prediction comes from the combined action of all base learners. Researchers have proposed three methods to handle comprehension of ensemble systems through instance-level attribution handling across all ensemble components and direct explanation of fusion/voting methods and construction of interpretable surrogate models based on ensemble outputs. The Bangla literature lacks systematic research, which would fulfill two objectives by demonstrating predictive capacity through ensemble-based explanations (Yeasmin et al., 2021).

The reviewed body of work has several important gaps which remain unaddressed. First, many studies continue to rely on small-scale, synthetic, crowd-sourced, or heavily preprocessed datasets that do not adequately reflect real-world Bangla news diversity, dialectal variation, code-mixing, orthographic inconsistency, or domain shift. The current evaluation protocols suffer from incompleteness because they fail to present essential practical aspects which include memory usage and inference speed and energy consumption and edge device deployment and especially interpretability. The decision-making process (how and why specific base models perform best on certain instances) needs to be examined through analysis and visualization whenever ensemble methods become operational. The Bangla news classification systems have no documented sociotechnical factors which they need to solve before entering service in Bangladesh West Bengal and throughout the worldwide Bangla diaspora (Rahman et al., 2021).

The present study directly tackles these limitations by conducting a comprehensive, multi-faceted empirical investigation. The study uses a large realistic dataset which contains 11904 manually labeled Bangla news articles from verified Bangladeshi media outlets (Hasan et al., 2023). The evaluation process tests six classical machine learning classifiers against four modern deep neural architectures while it uses specially designed ensemble methods to enhance predictive performance and it employs a standard post-hoc interpretability framework to create instance-level explanations. The research aims to develop dependable and trustworthy Bangla news classification systems through its dual pursuit of classification excellence and system transparency (Rahman, Khan, & Biswas, 2021).

3. Methodology

This part talks about all the different parts of the experiment that were used in the study. This relates to the collection of the data and the process of how it could be obtained. It further relates to the text processing chain for the text data, how features could be extracted, the machine learning model architecture and the deep learning model architecture, the techniques of ensemble learning, the application of explainability, the experiment setup, the criteria of experiment evaluation, and ensuring the reproduction of the experiment results.

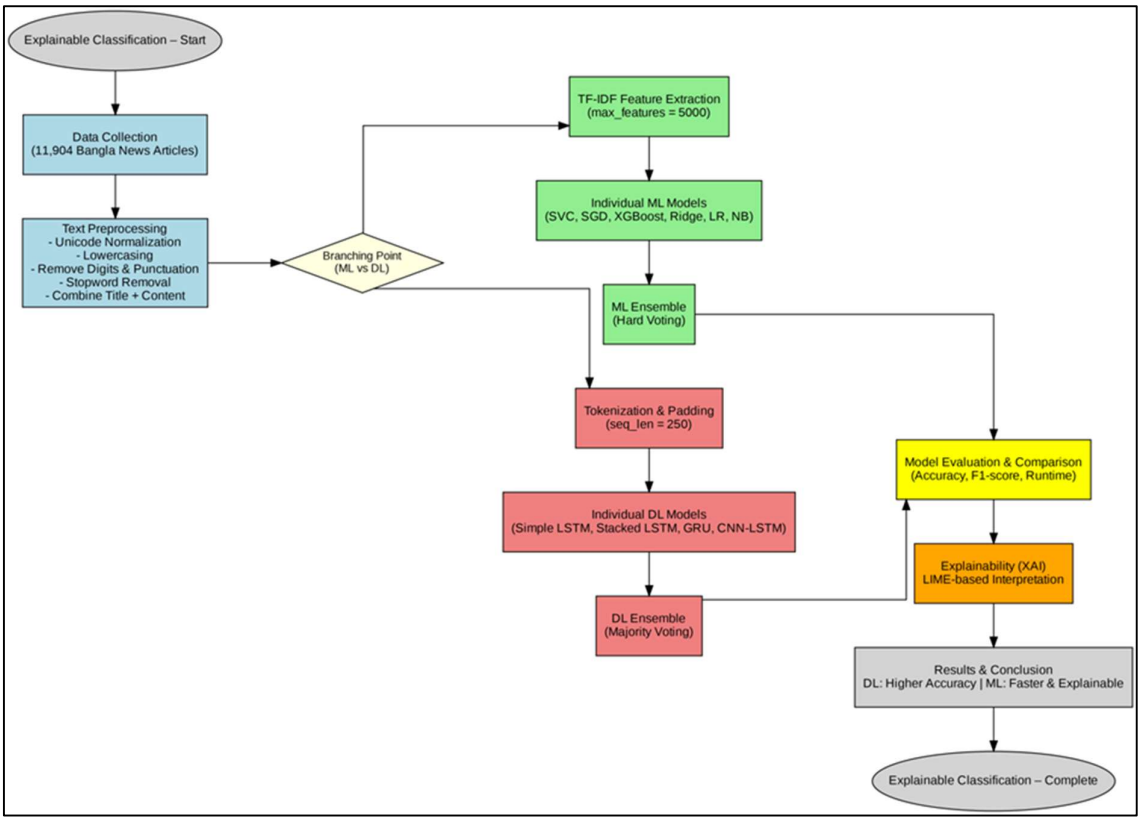

Figure 1. This figure illustrates the complete research workflow, from data collection and preprocessing to model training, ensemble methods, LIME explainability, and performance evaluation.

3.1. Dataset Description

The study makes use of a real Bangla news corpus titled "Over 11,500 Bangla News for NLP," which was sourced from the Kaggle repository. This dataset contains 11,904 news articles originally gathered from reliable news websites of Bangladeshi origin. Each entry includes the article title, publication date, list of reporters, category label, and the complete text of the news.

The articles are classified into four distinct categories: National, International, Sports, and Entertainment. The corpus consists of actual news pieces of varying sizes (ranging from short descriptions to lengthy news reporting). It reflects real-world data challenges, containing graphics, citations (omitted during processing), occasional grammatical errors, casual language, and discipline-specific terminology. To ensure data integrity, entries with missing reporter names were updated to 'Unknown,' making the dataset complete and accurate. The study relies entirely on these authentic Bangla news stories without using synthetic or external information sources (Ibrahim et al., 2021).

With approximately 12,000 samples, the dataset provides a substantial volume for training both traditional machine learning models and medium-sized deep neural networks. Its compact size ensures

suitability for language applications with limited storage. Furthermore, the class distribution is relatively balanced, eliminating the need for extensive resampling processes.

3.2. Text Preprocessing

Effective preprocessing is essential for Bangla text due to its script complexity, conjunct characters, and variable word forms. The pipeline consists of the following sequential steps:

1. Dealing with Missing Values: To prevent problems with processing, empty values in any field (which occurs rarely) are replaced with empty strings.

2. Unicode Normalization: This edition does not express this statement, but it is assumed that you will provide your text already converted into standard Unicode normalization form NFC.

3. Lowercasing: The entire text is converted to lowercase to reduce the vocabulary and ensure that the differences based on cases are balanced.

4. Removing Digits: The regular expression is utilized to remove Bangla digits (০–৯). It removes unwanted digits since they don't aid you in finding news.

5. Python's 'string.punctuation' set removes all the usual punctuation symbols, thereby making it impossible to use them as tokens.

6. White Space 'Normalization.Spaces' at the beginning and end of strings and multiple spaces between words are eliminated by removing leading spaces, trailing spaces, and reducing multiple spaces to a single space.

7. Stop-word Filtering: A list of 21 common Bangla stop-words (such as এ, ও, যে, সে, তা, এবং, করে, হয়, থেকে, যা, এর, এই, কি, না, তো, হতে, আর, আমি, তুমি, সম্পর্কে, জন্য) is used to get rid of tokens that are high-frequency but low-information.

After going through these steps for the content field, the cleaned content is placed together with the original title to produce the final input text. This way of putting the title and text together gets both the meaning of the headline and the information of the body.

3.3. Feature Extraction for Machine Learning

TF-IDF vectorization changes the combined text into numbers that normal machine learning models can use. The vectorizer has a limit of 5,000 features so that it can find a fair balance between speed and expressiveness. Because word-level unigrams with TF-IDF have worked effectively for Bangla news in the past, there are no extra n-gram features or custom tokenizers used. The generated sparse matrix is split into training (70%) and testing (30%) sets using a fixed random seed for reproducibility (Ikonomakis et al., 2005).

3.4. Models for Machine Learning

There are six classifiers that use different kinds of algorithms:

- Logistic Regression - It is a linear probabilistic model with L2 regularization with a possible number of repetitions of 500.

- LinearSVC is a SVM algorithm that is suited to text input with more than one dimension.

- Ridge Classifier: A linear model that treats classification as regression by adding an L2 penalty.

- SGD Classifier: This is a stochastic gradient descent model based on logistic loss that can work with more than one class.

- Multinomial Naive Bayes is a probabilistic model that works well with text counts and TF-IDF features.

- XGBoost is a tree-based gradient boosting framework that has 300 estimators, a maximum depth of 6, and a learning rate of 0.1.

All models are trained on the same TF-IDF training set with the default hyperparameters, except for some that are changed to make them more stable (for example, by making it take longer to reach convergence).

3.5. Deep Learning Models

The Keras Tokenizer breaks up the joined text into tokens that can be used in the deep learning pipeline. It only works with the 10,000 most common words. There are always 250 tokens in a sequence, so they are either taken away or added to. Labels in a 4D category format use one-hot encoding. The training and testing sets are split into two groups using the same random seed. 20% of the data is used for testing, and 80% is used for training (Liang & Yi, 2021).
Four neural architectures are developed:

- Basic LSTM: This is a single-layer LSTM with 64 units and a softmax output that is very full.

- Stacked LSTM: There are two LSTM layers. There are 64 units in the first one and 32 units in the second one.

- GRU: To save space, there is only one Gated Recurrent Unit layer with 64 units.

- CNN-LSTM Hybrid: A Conv1D layer with 64 filters and a kernel size of 3, followed by max-pooling, and then an LSTM layer with 64 units.

The embedding layer in all of the models has 10,000 input dimensions and 64 output dimensions. We also use the Adam optimizer, categorical cross-entropy loss, and a batch size of 128. Training is limited to three epochs to prevent overfitting on the small dataset while monitoring the model's convergence.

3.6. Ensemble Strategies

For each paradigm, there is a different way to use an ensemble:

- ML Ensemble: To get the most out of the four best linear models (LinearSVC, Ridge Classifier, Logistic Regression, and SGD Classifier), we use hard voting.
- DL Ensemble: The three deep models that work best (Simple LSTM, CNN-LSTM, and Stacked LSTM) vote on which one is the best. You can make predictions by taking the mode of each model's output (Mohammed & Kora, 2022).
  These groups strive to bring together diversity and stability to make it easier to generalize.

3.7. Explainable AI Implementation

For two common ML models, the Ridge Classifier (a linear high-performer) and the Multinomial Naive Bayes (a probabilistic baseline), we apply Local Interpretable Model-agnostic Explanations (LIME). To explain single cases, LIME changes the input text and fits a linear surrogate model that can be comprehended in the area where it is used (Cesarini et al., 2024). The explanation reveals the 10 words or phrases that have the most impact, good or bad, on each predicted class. There is an HTML output with a white background for explanations, which makes them simple to see. They are also stored as files for more detail (Madi et al., 2024).

3.8. Experimental Setup and Evaluation

We use Python 3.x for all of our tests, LIME for explainability, TensorFlow/Keras for DL, and scikit-learn for ML. If you use a random seed of 42 for the whole world, you can do everything again. GPUs do not need to speed up machine learning models. Deep learning models can run on CPUs, but they work best on GPUs like the Tesla P100. Some ways to measure are:

- Correctness
- F1-score, recall, and precision with weights
- Matrices of confusion
- Time to learn and make guesses
- Learning curves for models that use deep learning
- Visual LIME explanations to help you understand things better in a qualitative way

The groups are about the same size, so stratified division is not always followed. But random seed consistency guarantees that the outcomes are always the same.

3.9. Problems with Reproducibility and Ethics

The whole pipeline is easy to predict because the seeds are set. The dataset comes from public sources, and any personal information has been removed. There is no information stored or processed that can be used to identify a person. We test all models on test sets that have not been used before to prevent data leakage. The research follows the guidelines for responsible AI by focusing on clarity and openness (Onan et al., 2016).

This method offers a strong, repeatable way to test how well Bangla news classification works and how easy it is to understand.

4. Results and Analysis

This part shows all the test results from the evaluation process. We provide a comprehensive analysis of quantitative performance indicators for each classifier, various ensemble configurations, runtime comparisons, insights derived from confusion matrices, trends observed in learning curves, and qualitative interpretability outcomes from LIME explanations. The results come from the test set that was held back. This is to make sure that the test for generalization is fair.

4.1. Performance of Individual Machine Learning Models

When trained on the TF-IDF representation of the combined title and content features, the classical machine learning classifiers were able to make good predictions. The Linear Support Vector Machine (LinearSVC) was the best of the six models tested, with an accuracy rate of 92.36%. The Ridge Classifier was not far behind, at 92.22%. The SGD Classifier and Logistic Regression both earned 91.94%, while the XGBoost got 90.99%. Multinomial Naive Bayes came in last with a score of 90.40%. This means that its strong assumption of independence does not work well with news stories, which often feature text patterns that are very related.

The confusion matrix in Figure 2 shows the classification performance of the best single ML model (LinearSVC, 92.36% accuracy), highlighting correct predictions and errors across the four classes.

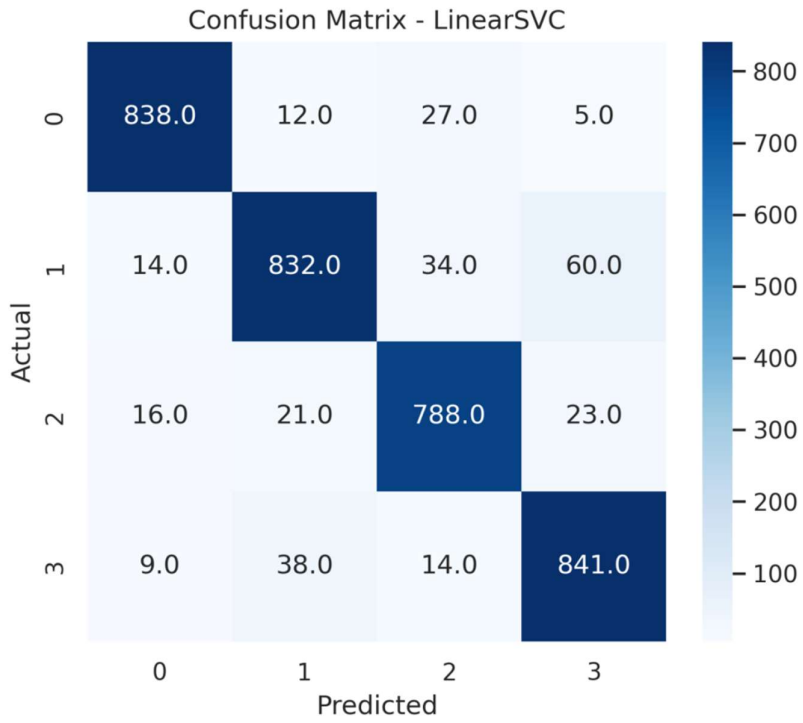

Figure 2. Confusion Matrix of LinearSVC (Best Individual ML Model – 92.36% Accuracy).

Figure 3 shows a chart comparing the key metrics (accuracy, precision, recall, F1-score) of the LinearSVC model, demonstrating balanced performance around 92.3–92.4%.

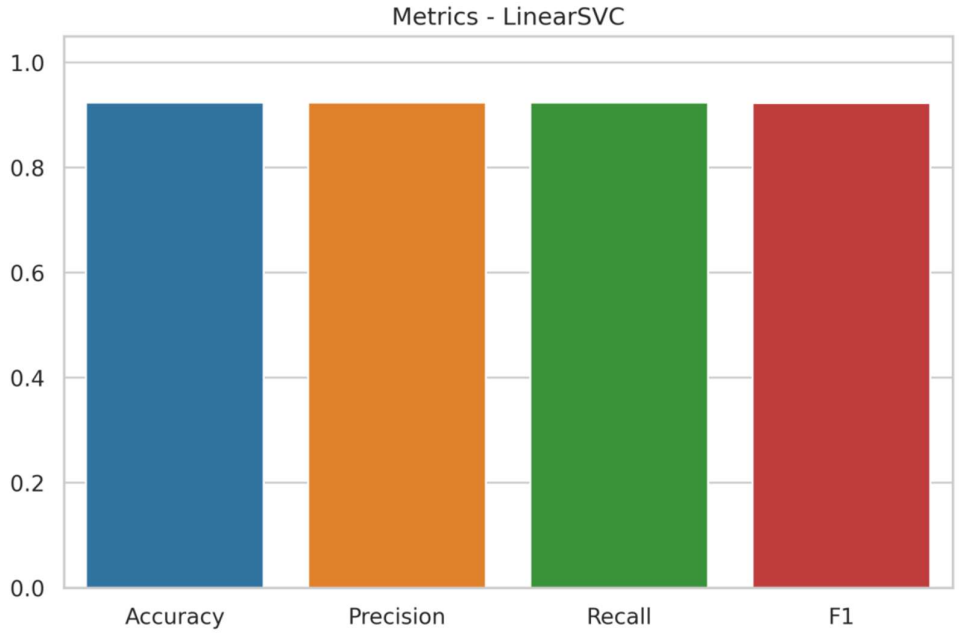

Figure 3. Performance Metrics of LinearSVC (Accuracy, Precision, Recall, F1-Score).

The weighted precision, recall, and F1-scores were all extremely close to the accuracy in all of the models. The best balance across the four classes was with LinearSVC, which had 92.38% precision, 92.36% recall, and 92.35% F1-score. The linear models (LinearSVC, Ridge, Logistic Regression, and SGD) always did better than the tree-based and probabilistic models. This means that they work well with text data that is sparse and has a number of dimensions. XGBoost has a more sophisticated gradient boosting framework, but it did not work better than the basic linear models. This suggests that adding more steps did not assist in this case. The metrics for Multinomial Naive Bayes were the worst, perhaps because it did not explain how features worked together very well.

Table 1 summarizes the accuracy, weighted precision, recall, F1-score, and runtime of the six classical machine learning models evaluated on the Bangla news test set. Tests done while the computers were running showed that they worked considerably differently. The fastest option was Multinomial Naive Bayes, which finished training and inference in less than 0.01 seconds. It took roughly 0.14 seconds for the Ridge Classifier and LinearSVC to finish, and about 0.07 seconds for the SGD Classifier. It took 1.90 seconds for Logistic Regression to work since it had to go through a process of optimization that happened over and over. XGBoost took the longest, 58.66 seconds, because it had to make a lot of trees and adjust the slopes at the same time.

Table 1. Performance Comparison of Individual Machine Learning Classifiers

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Runtime (sec)
Logistic Regression	91.94	92.02	91.94	91.94	1.90
LinearSVC	92.36	92.38	92.36	92.35	0.14
Ridge Classifier	92.22	92.29	92.22	92.21	0.13
SGD Classifier	91.94	92.01	91.94	91.94	0.07
Multinomial Naive Bayes	90.40	90.60	90.40	90.43	0.01
XGBoost	90.99	91.05	90.99	90.98	58.66

4.2. Machine Learning Ensemble Performance

We constructed a hard-voting ensemble by putting together the Ridge Classifier, Logistic Regression, and SGD Classifier, which are the three best linear classifiers. The total performance of this group was the best, with an accuracy of 92.55%, a precision of 92.58%, a recall of 92.55%, and an F1-score of 92.55%. The fact that it got better by about 0.19 percentage points above the best single model (LinearSVC) shows how useful it is to integrate different decision limitations, especially when the models disagreed on cases that were on the edge or unclear. The ensemble performed a wonderful job of retaining the class balance, and the whole prediction phase only took 1.69 seconds. This shows that linear model ensembles are still useful even when there are multiple predictors.

Figure 4 displays the improved performance of the ML hard-voting ensemble (92.55% accuracy), with reduced misclassifications compared to individual models. Meanwhile, Figure 5 presents the accuracy, precision, recall, and F1-score of the ML ensemble, showing a small but consistent gain over the best single ML model.

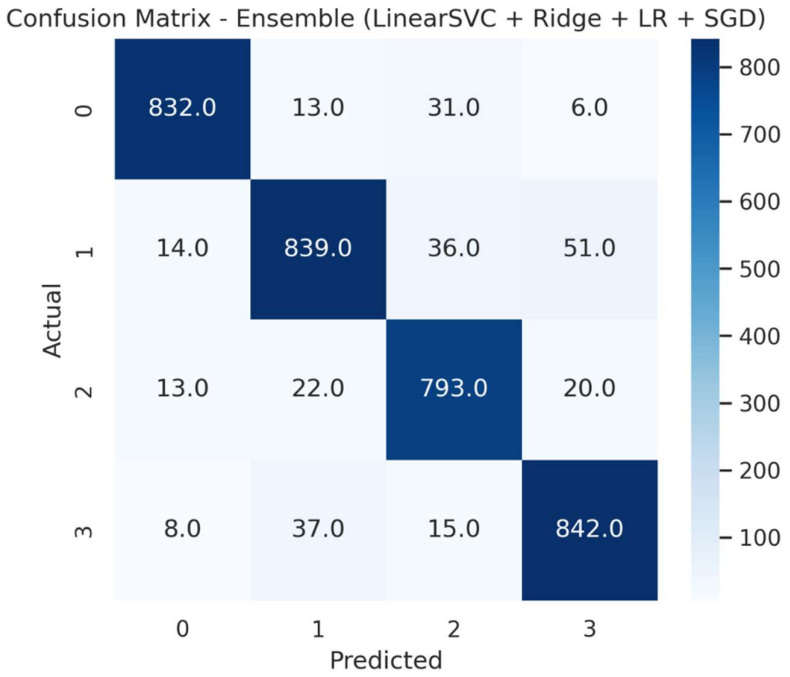

Figure 4. Confusion Matrix of ML Ensemble Using Hard Voting (92.55% Accuracy).

The confusion matrix analysis of the ML ensemble showed that the most common mistakes were made between the International and National categories. This is probably because both groups use the same kinds of geopolitical terminology and ideas. The categorization rates for Sports and Entertainment goods were almost perfect since the words used to describe them were very different.

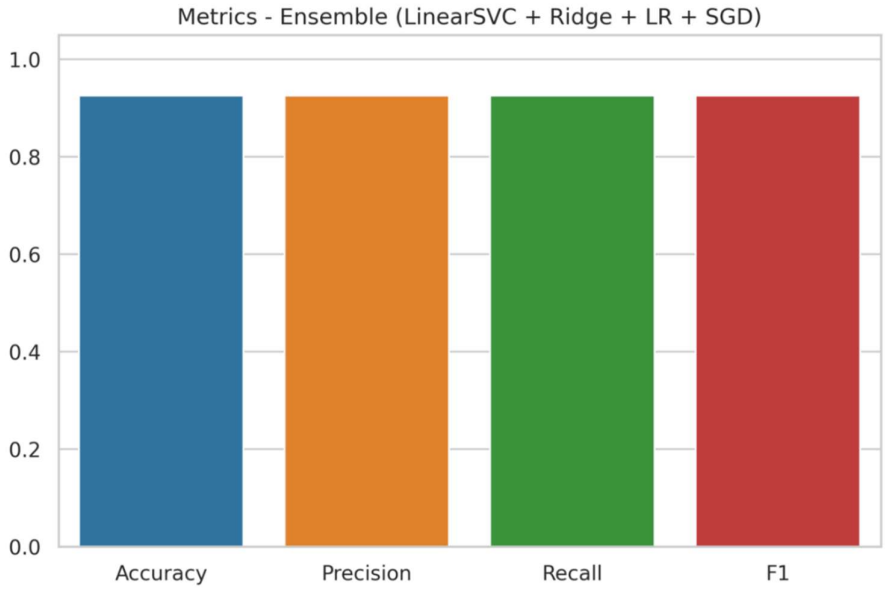

Figure 5. Performance Metrics of the ML Ensemble Model.

4.3. Performance of Individual Deep Learning Models

The deep neural architectures always did better than the machine learning models when it came to making predictions. The Simple LSTM was the best single model, with an accuracy of 95.46%, a weighted precision of 95.50%, a recall of 95.46%, and an F1-score of 95.46%. The CNN-LSTM hybrid came in second with 95.42%, which indicates how useful it is to combine convolutional feature extraction with recurrent sequence modeling. The GRU obtained 94.79% and the Stacked LSTM only got 93.53%. This might be because the extra layer that keeps coming back made it more likely to fit too well. Figure 6 shows training and validation accuracy curves over epochs for the Simple LSTM model, indicating smooth convergence without significant overfitting.

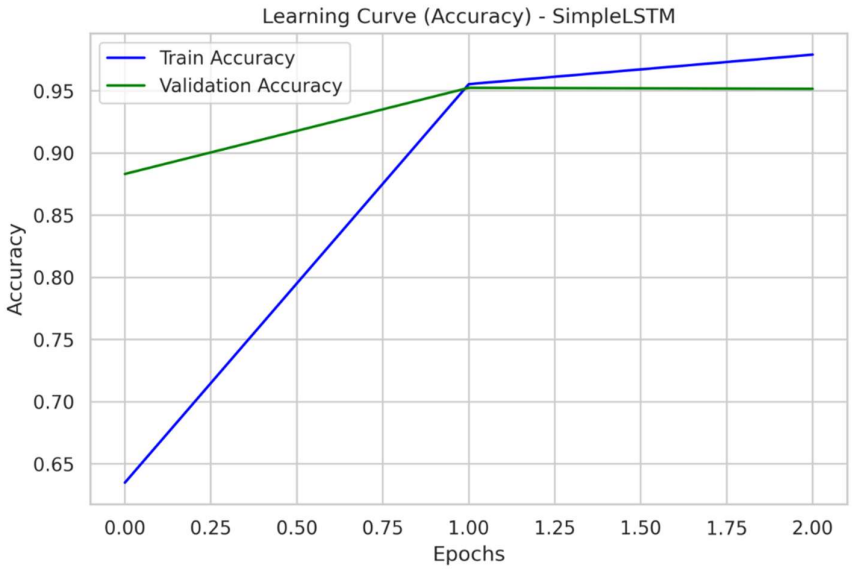

Figure 6. Learning Curves Accuracy of Simple LSTM (95.46% Accuracy).

Figure 7 illustrates the decreasing training and validation loss over epochs for the Simple LSTM model, confirming stable convergence.

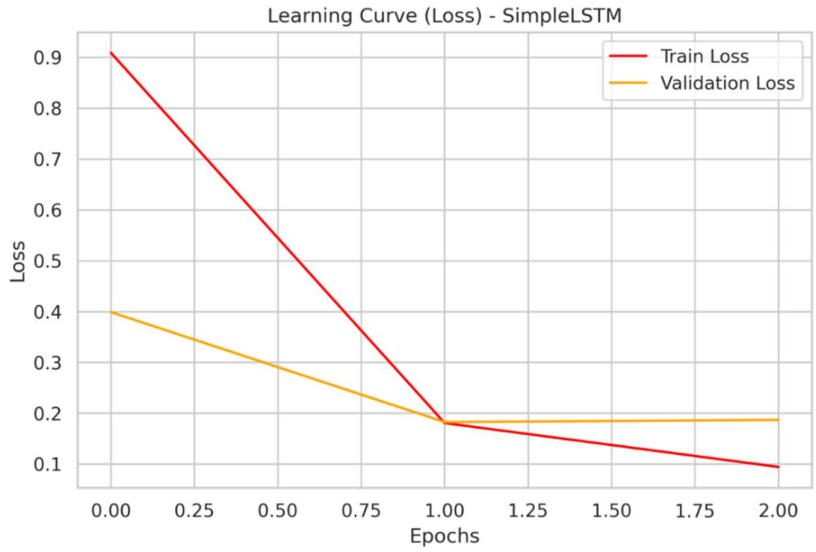

Figure 7. Learning Curves Loss of Simple LSTM (95.46% Accuracy).

Table 2 reports the accuracy, weighted precision, recall, and F1-score achieved by the four deep learning models (Simple LSTM, Stacked LSTM, GRU, CNN-LSTM Hybrid) on the Bangla news classification task. All of the deep models did well for each class, and the weighted metrics were quite close to accuracy. This means that there were not many difficulties with imbalance. The training only took three epochs, but the validation curves showed that they came together quickly. At the end of the last epoch, the Simple LSTM had a training accuracy of 96.83% and a validation accuracy of 95.46%. There was a clear drop in the validation loss and a clear rise in the validation accuracy. This means that the model was learning slowly and not getting too good at it.

Table 2. Performance Metrics of Individual Deep Learning Architecture

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Simple LSTM	95.46	95.50	95.46	95.46
Stacked LSTM	93.53	93.51	93.53	93.51
GRU	94.79	94.80	94.79	94.77
CNN–LSTM Hybrid	95.42	95.43	95.42	95.42

4.4. Deep Learning Ensemble Performance

The best result from the whole study came from the majority vote of the three best deep models: Simple LSTM, CNN-LSTM, and Stacked LSTM. The accuracy was 95.76%, the precision was 95.77%, the recall was 95.76%, and the F1-score was 95.75%. This small improvement over the best single deep model shows that using a mix of models in an ensemble can help make predictions that are more accurate and work in more situations. Even though it had to go via three different networks, the ensemble still required a fair length of time to make a choice, about 1.60 seconds.

Figure 8 depicts the superior classification results of the DL majority-voting ensemble (95.76% accuracy), with minimal errors across all categories. Figure 9 summarizes the top performance of the DL ensemble (95.76% accuracy), highlighting its advantage over ML approaches.

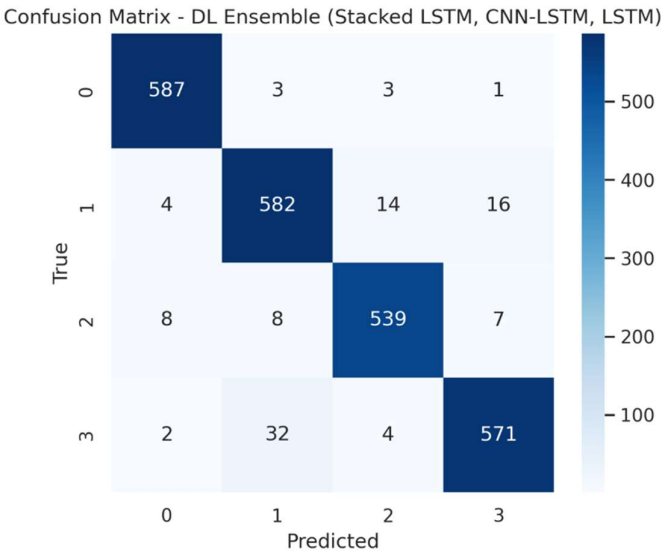

Figure 8. Confusion Matrix of DL Ensemble with Majority Voting (95.76% Accuracy).

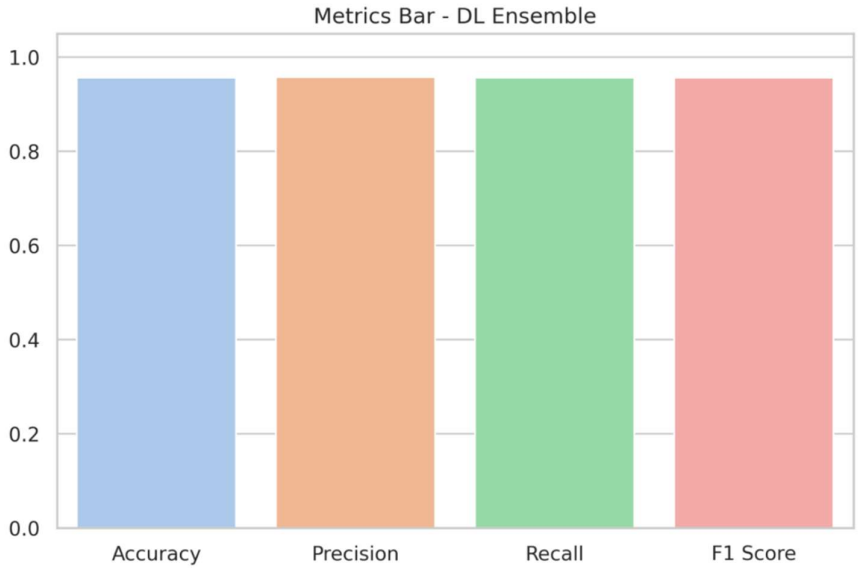

Figure 9. Performance Metrics of the Deep Learning Ensemble.

4.5. Comparative Performance Summary

The ensemble of deep learning got the most right: 95.76%. This was 3.21 percentage points better than the best ML group. This comparison shows that neural networks are better than other types of architectures at finding both sequential and contextual dependencies in Bangla news articles. The ML ensemble, on the other hand, was far better at math since it learnt and made predictions much faster than deep models. It was easy to understand how the entire ML process worked. But it was harder to see deep models after the incident (Li et al., 2022). Table 3 compares the predictive performance (accuracy, precision, recall, F1-score) and inference runtime of the best ML ensemble (hard voting) and DL ensemble (majority voting).

Table 3. Comparative Performance of Machine Learning and Deep Learning Ensembles

Ensemble Type	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Runtime (sec)
ML Ensemble (Hard Voting)	92.55	92.58	92.55	92.55	1.69
DL Ensemble (Majority Voting)	95.76	95.77	95.76	95.75	1.60

4.6. Qualitative Insights from LIME Explanations

We built LIME explanations for the Multinomial Naive Bayes and the Ridge Classifier to help users understand two common ML models better on a case-by-case basis. Both models showed that for Entertainment articles that were accurately detected, the most relevant positive criteria were domain-specific keywords including star names, movie titles, release dates, and media-related terms. These explanations were very similar to what people would expect, which made it seem like the models used semantically relevant signals instead of deceptive connections (Tiwari, 2024).

Figure 10 displays the LIME visualization highlighting influential words for a correctly classified Entertainment article using the Ridge Classifier, showing reliance on relevant domain-specific terms.

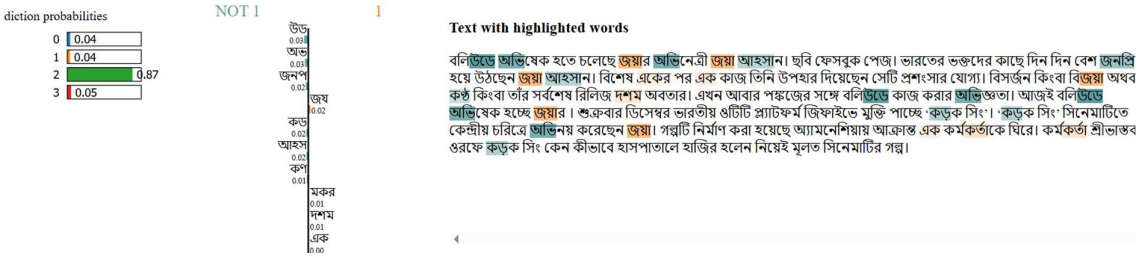

Figure 10. LIME Explanation for a Correctly Classified Sample (Entertainment – Ridge Classifier).

The LIME visualization in Figure 11 reveals feature contributions for a misclassified National/International case, exposing over-reliance on overlapping surface-level terms.

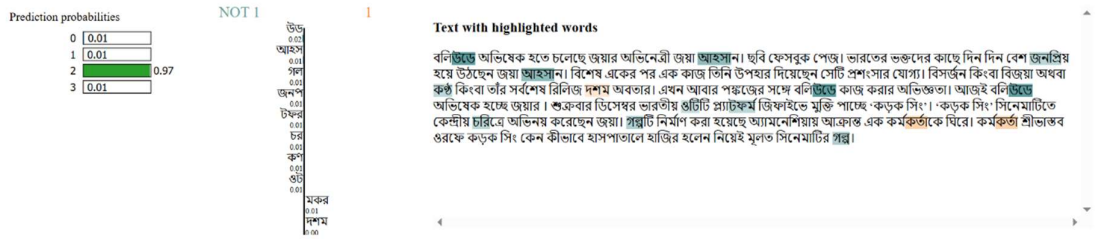

Figure 11. LIME Explanation for a Misclassified / Borderline Case (National vs. International).

LIME showed that people relied too much on surface-level features, like geographical names that were in both International and National publications, in cases that were not obvious or possibly misclassified (Gurrapu et al., 2023). The weights of a regularized linear model were expected to be smoother and more evenly spread out than those of Multinomial Naive Bayes. These visual and numerical insights make consumers more likely to trust the system and demonstrate where it needs to be improved in the future, including how to better handle overlapping domain terminology (Mathews, 2019).

4.7. Key Observations and Implications

The results show that performance and efficiency are in sync. Deep learning ensembles are the best choice if you need the most accurate predictions (95.76%) and have ample computing power. The ML ensemble (92.55%) works almost as well as the other models, but it uses less power and is easier to understand. It is considerably easier to understand old models with LIME explanations. This lets those who are interested in the project go over the choices and find any problems that might come up (Kamath et al., 2018).

These results demonstrate that the ideal way to sort Bangla news in the real world would be to utilize hybrid deployment strategies. These use quick ML models for the first high-throughput filtering and deep ensembles for decision making. Post-hoc interpretability makes sure that both expected accuracy and trustworthiness in people are taken care of.

5. Discussion

The above studies from the previous sections revealed interesting trends and effects related to the compromises between the speeds, accuracy, and understandability regarding the sorting of Bangla news. This section addresses the major points, places the study within the larger context of language processing for low-resource languages, highlights the flaws with the current study, and proposes ways to do better in the future.

The deep learning ensemble had the highest overall accuracy of 95.76%. This is an improvement of 3.21% above the top performing set of machine learning models. This result is consistent with what has long been known regarding how structures in the human brain facilitate the detection of complex sequential, contextual, and hierarchical patterns that are inherent in natural language. The Simple LSTM model attained an accuracy of 95.46%, followed by the CNN-LSTM hybrid model, which achieved 95.42%. That means even a basic kind of recurrent and hybrid models can learn to represent Bangla news content accurately on being trained on a medium-sized dataset. The use of majority voting in this ensemble reduced the gap between the clusters and helped the generalization. This means that with different sorts of architectures, such as convolutional-recurrent and recurrent, a system is much more stable (Chattaraj & Chimalakonda, 2025). Table 4 highlights the trade-offs in training time, prediction time, resource requirements, and key advantages between ML ensembles, individual DL models, and DL ensembles.

Table 4. Computational Efficiency and Resource Usage Trade-offs.

Model / Ensemble	Training Time (sec)	Prediction Time (sec)	Resource Usage	Key Advantage
ML Ensemble	~2–60	1.69	Low (CPU)	Fast deployment and interpretability
DL Individual (e.g., Simple LSTM)	~8–10 per epoch	0.68–1.02	High (GPU)	High predictive accuracy
DL Ensemble	N/A	1.60	High	Best overall performance

On the other hand, the mean machine learning ensemble got 92.55% of the questions correct with minimal cost. The most regularly used types of machine learning methods in this competition were linear models, including the LinearSVC, Ridge Classifier, Logistic Regression, and SGD Classifier. This shows that TF-IDF representations still work well for Bangla text data that is sparse and has a lot of dimensions. It is a little easier to understand when you put these linear predictors together. This means that if you have different decision constraints, you can get better results without utilizing more complicated tree-based or neural methods. The linear family outperformed both XGBoost and Multinomial Naive Bayes. This means that adding more features to the model might not help in this case.

One of the most significant things we learnt is that being efficient and being good at something are not the same thing. Deep ensembles take a long time to learn and make predictions (a few seconds each epoch on GPU). On a CPU, the ML ensemble can do it in less than two seconds. This distinction is quite important when utilizing it in real life, as in mobile apps, local news aggregators, or places with slow internet, which is common in Bangladesh and nearby areas. If you need to quickly look at a lot of data, like filtering news in real time or censoring social media posts, the ML pipeline is the ideal way to do it. It does not lose much of its predicitve ability (Yuan et al., 2024).

Another important thing to think about is how easy it is to get. You can easily understand traditional linear models because the weights of the features show how important each word is, and you can check the predictions by hand. LIME explanations make this extra clearer by showing how models use specific keywords from a category, like actor names for entertainment and geopolitical ideas for international, to construct classifications that are correct at the instance level. These explanations also bring up problems, such relying too much on surface cues (like location names that mix up International and National categories), which makes it clear what needs to happen next (Suneera & Prakash, 2020).

Deep models are more accurate, but they are hard to understand. Attention approaches may yield restricted insights, although they frequently do not generate logical arguments for people. The current study intentionally directed LIME towards ML models to underscore its transparency benefit. This alternative makes a valid point: where explainability is important (as when following rules, checking for bias, or using media tools that the public can see), machine learning ensembles with post-hoc explanations might be the best choice.

A step further is the analysis of the pattern of misclassification that was observed. The International and National categories being grouped into one were probably due to their heavy use of common words, such as country names, diplomatic talk, and political personalities. On the other hand, news articles with regards to sports and entertainment were close to correct classification, using less common words. What follows from the results, therefore, is that future research might include domain adaptive feature learning or sub-models for the better classification of similar class labels.

When looked at from a broader perspective, the results demonstrate that hybrid approaches work effectively for languages with fewer resources. Deep learning is still getting better at being accurate, but classical ML is still very beneficial when there is less data, few computers, or when it is more vital to be able to interpret the results. Adding explanations like those in LIME-style makes the difference between how well something works and how much you can trust it smaller. This is a very important need in Bangla NLP, where automated technologies are changing the way people share information in a big way.

There are several considerations. Firstly, the dataset is realistic and has a lot of articles (11,904), although it is not as big as the ones that language researchers with a lot of resources usually use. This restriction makes deep models less useful because they usually work better with a lot more data. Second, preprocessing was kept simple (only deleting stop words, no stemming or lemmatization) so that the results could be repeated and the main goal could be to compare models. It would be even better if there were more complex tools that just worked with Bangla. Third, we only trained the deep models for three epochs so they would not fit very well. More training with early stopping or regularization might help even more. Lastly, only ML models used LIME to explain things. It is still feasible to make deep ensembles easier to understand after the fact. Such constraints will also provide us insight into how to proceed in a positive manner. Deep networks may significantly outperform if there can be additions of pretrained language models that are specific to the Bangla language, such as BanglaBERT. Sophisticated preprocessing methods, including rule-based methods for stemming and lemmatization along with named entity recognition, may contribute to noise reduction and boosting the generalization capability of the proposed model. An appropriate comparison of different methods of XAI may provide further insights apart from the proposed approaches, which may include methods like SHAP and Integrated gradients for deep networks.

This paper demonstrates the capability of achieving high accuracy, efficient computation, and simple interpretations concurrently while dealing with news categorization of the Bangla language. The deep ensemble model performs the best and is the most suitable one for prediction purposes, while the ML model with the help of LIME interpretations is the most convenient one. All of these pieces provide a

way to create systems that are flexible, reliable, and adaptable to various digital environments where the Bangla language is spoken.

6. Conclusion and Future Work

This study solved major obstacles which Bangla natural language processing faces during multiclass news classification in low-resource environments. This was done by solving two main challenges which required high accuracy predictions together with transparent operations and efficient resources and real-world operational capacity. A complete empirical assessment of machine learning methods was conducted against deep neural networks through the evaluation of 11,904 Bangla news articles collected from authentic Bangladeshi news sources. The news articles were collected using classical machine learning classifiers and deep neural network methods ensemble techniques, while using LIME post-hoc interpretability on certain ML models.

The results show that deep learning ensembles achieve the best classification results through their ability to capture sequenced and context-specific patterns from Bangla text. At the same time, machine learning ensembles achieve results which approach competitive levels while needing fewer resources and delivering faster CPU processing times and providing LIME explanations which show the significance of features and their associated limitations by showing when models depend too much on similar domain terms.

The framework achieves its purpose by delivering accurate predictions through efficient operations which maintain system transparency, thereby creating trustworthy Bangla news classification systems. The hybrid approach meets social accountability standards for operating news aggregation platforms and misinformation detectors and content recommendation engines and automated media monitoring tools in Bangladesh and West Bengal and among Bangladeshi speakers who live abroad.

The current dataset contains realistic substantial size but it falls short of the dataset size from high-resource languages which limits deep model generalization abilities. The researchers used simple preprocessing techniques to achieve reproducibility while LIME application focused on ML models as the main target. The researchers plan to improve performance through pre-training Bangla language models which include BanglaBERT and its variants which have shown success in text classification. The researchers plan to develop an advanced system through three core areas which include creating advanced Bangla-aware preprocessing capabilities which consist of stemming and lemmatization and named entity recognition and creating new XAI techniques which will join the existing techniques of SHAP and Integrated Gradients for deep ensembles while testing the system through cross-domain transfer learning and bias analysis and adversarial robustness testing and user studies. This study shows that a special combination of high accuracy and efficiency, together with interpretability produces complementary results which scientists can use to create secure and socially responsible Bangla NLP systems that better serve digital information systems in low-resource areas.

References

Ahmad, I., AlQurashi, F., & Mehmood, R. (2022). Machine and deep learning methods with manual and automatic labelling for news classification in bangla language. *arXiv preprint arXiv:2210.10903*. doi:10.48550/arXiv.2210.10903

Alam, F., Hasan, A., Alam, T., Khan, A., Tajrin, J., Khan, N., & Chowdhury, S. A. (2021). A review of bangla natural language processing tasks and the utility of transformer models. *arXiv preprint arXiv:2107.03844*. https://doi.org/10.48550/arXiv.2107.03844

Atanasova, P., Simonsen, J. G., Lioma, C., & Augenstein, I. (2020, November). A diagnostic study of explainability techniques for text classification. In *Proceedings of the 2020 conference on empirical*

methods in natural language processing (EMNLP) (pp. 3256-3274). https://doi.org/10.1007/978-3-031-51518-7_7

Banik, N., Rahman, M. H. H., Chakraborty, S., Seddiqui, H., & Azim, M. A. (2019, May). Survey on text-based sentiment analysis of bengali language. In *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)* (pp. 1-6). IEEE. https://doi.org/10.1109/ICASERT.2019.8934481.

Cesarini, M., Malandri, L., Pallucchini, F., Seveso, A., & Xing, F. (2024). Explainable ai for text classification: Lessons from a comprehensive evaluation of post hoc methods. *Cognitive Computation*, *16*(6), 3077-3095. https://doi.org/10.1007/s12559-024-10325-w

Chakraborty, S., Das, P., Dipto, S. M., Pramanik, M. A., & Noor, J. (2025). An Analytical Review of Preprocessing Techniques in Bengali Natural Language Processing. *IEEE Access*. https://doi.org/10.1109/ACCESS.2025.3574234

Chattaraj, R., & Chimalakonda, S. (2025). NLP Libraries, Energy Consumption and Runtime: An Empirical Study. *Proceedings of the ACM on Software Engineering*, *2*(FSE), 2850-2873. https://doi.org/10.1145/3729396

Emon, E. A., Rahman, S., Banarjee, J., Das, A. K., & Mittra, T. (2019, June). A deep learning approach to detect abusive bengali text. In *2019 7th International Conference on Smart Computing & Communications (ICSCC)* (pp. 1-5). IEEE. https://doi.org/10.1109/ICSCC.2019.8843606

Gohel, P., Singh, P., & Mohanty, M. (2021). Explainable AI: current status and future directions. *arXiv preprint arXiv:2107.07045*.

Gurrapu, S., Kulkarni, A., Huang, L., Lourentzou, I., & Batarseh, F. A. (2023). Rationalization for explainable NLP: a survey. *Frontiers in artificial intelligence*, *6*, 1225093. https://doi.org/10.3389/frai.2023.1225093

Hasan, M. K., Islam, S. A., Ejaz, M. S., Alam, M. M., Mahmud, N., & Rafin, T. A. (2023). Classifying bengali newspaper headlines with advanced deep learning models: Lstm, bi-lstm, and bi-gru approaches. *Asian Journal of Research in Computer Science*, *16*(4), 372-388. https://doi.org/10.9734/ajrcos/2023/v16i4398

Hossain, M. R., Sarkar, S., & Rahman, M. (2020). Different machine learning based approaches of baseline and deep learning models for bengali news categorization. *International Journal of Computer Applications*, *975*, 8887. https://doi.org/10.5120/ijca2020920107

Hossain, T., Islam, A. R., Mehedi, M. H. K., Rasel, A. A., Abdullah-AL-Wadud, M., & Uddin, J. (2025). BanglaNewsClassifier: A machine learning approach for news classification in Bangla Newspapers using hybrid stacking classifiers. *PLoS One*, *20*(6), e0321291. https://doi.org/10.1371/journal.pone.0332710

Ibrahim, Y., Okafor, E., Yahaya, B., Yusuf, S. M., Abubakar, Z. M., & Bagaye, U. Y. (2021, July). Comparative study of ensemble learning techniques for text classification. In *2021 1st International Conference on Multidisciplinary Engineering and Applied Science (ICMEAS)* (pp. 1-5). IEEE. https://doi.org/10.1109/ICMEAS52683.2021.9692306

Ikonomakis, M., Kotsiantis, S., & Tampakas, V. (2005). Text classification using machine learning techniques. *WSEAS transactions on computers*, *4*(8), 966-974.

Kamath, C. N., Bukhari, S. S., & Dengel, A. (2018, August). Comparative study between traditional machine learning and deep learning approaches for text classification. In *Proceedings of the ACM Symposium on Document Engineering 2018* (pp. 1-11). https://doi.org/10.1145/3209280.3209526

Khan, Z. (2025). Natural Language Processing Techniques for Automated Content Moderation. *International Journal of Web of Multidisciplinary Studies*, *2*(2), 21-27. https://doi.org/10.13140/RG.2.2.34900.99200

Li, Q., Peng, H., Li, J., Xia, C., Yang, R., Sun, L., ... & He, L. (2022). A survey on text classification: From traditional to deep learning. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *13*(2), 1-41. https://doi.org/10.1145/3495162

Liang, D., & Yi, B. (2021). Two-stage three-way enhanced technique for ensemble learning in inclusive policy text classification. *Information Sciences*, *547*, 271-288.

Limon, S. M., Ahmad, M., & Mishu, F. N. (2018). Bangla News Classification Using Machine Learning. https://doi.org/10.1016/j.ins.2020.08.051

Madi, I. A. E., Redjdal, A., Bouaud, J., & Seroussi, B. (2024). Exploring explainable AI techniques for text classification in healthcare: a scoping review. *Digital Health and Informatics Innovations for Sustainable Health Care Systems*. https://doi.org/10.3233/SHTI240544

Mahoney, C. J., Zhang, J., Huber-Fliflet, N., Gronvall, P., & Zhao, H. (2019, December). A framework for explainable text classification in legal document review. In *2019 IEEE International Conference on Big Data (Big Data)* (pp. 1858-1867). IEEE. https://doi.org/10.1109/BigData47090.2019.9005659

Mandal, A. K., & Sen, R. (2014). Supervised learning methods for bangla web document categorization. *arXiv preprint arXiv:1410.2045*.

Mathews, S. M. (2019, July). Explainable artificial intelligence applications in NLP, biomedical, and malware classification: a literature review. In *Intelligent computing-proceedings of the computing conference* (pp. 1269-1292). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-22868-2_90

Mohammed, A., & Kora, R. (2022). An effective ensemble deep learning framework for text classification. *Journal of King Saud University-Computer and Information Sciences*, *34*(10), 8825-8837. https://doi.org/10.1016/j.jksuci.2021.11.001

Mustafa, S., & Hama Saeed, M. (2025). Empowering text classification with NLP and explainable AI for enhanced interpretability. *Journal of Electrical Systems and Information Technology*, *12*(1), 81. https://doi.org/10.1186/s43067-025-00273-2

Onan, A., Korukoğlu, S., & Bulut, H. (2016). Ensemble of keyword extraction methods and classifiers in text classification. *Expert Systems with Applications*, *57*, 232-247. https://doi.org/10.1016/j.eswa.2016.03.045

PRAMANIK, M. A., & NOOR, J. (2025). An Analytical Review of Preprocessing Techniques in Bengali Natural Language Processing. https://doi.org/10.1109/ACCESS.2025.3574234

Rahman, M. M., Khan, M. A. Z., & Biswas, A. A. (2021, January). Bangla news classification using graph convolutional networks. In *2021 International Conference on Computer Communication and Informatics (ICCCI)* (pp. 1-5). IEEE. https://doi.org/10.1109/ICCCI50826.2021.9402567

Rahman, S., Mithila, S. K., Akther, A., & Alam, K. M. (2021, July). An empirical study of machine learning-based Bangla news classification methods. In *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)* (pp. 1-6). IEEE. https://doi.org/10.1109/ICCCNT51525.2021.9579655

Roy, A., Sarkar, K., & Mandal, C. K. (2023). Bengali text classification: A new multi-class dataset and performance evaluation of machine learning and deep learning models. https://doi.org/10.21203/rs.3.rs-3129157/v1

Salehin, K., Ahmed, F., Nabi, M. A., & Alam, M. K. (2021). *Bangla text classification using machine learning and deep learning techniques* (Doctoral dissertation, Brac University).

Salehin, K., Alam, M. K., Nabi, M. A., Ahmed, F., & Ashraf, F. B. (2021, December). A comparative study of different text classification approaches for bangla news classification. In *2021 24th International Conference on Computer and Information Technology (ICCIT)* (pp. 1-6). IEEE. https://doi.org/10.1109/ICCIT54785.2021.9689843

Sen, O., Fuad, M., Islam, M. N., Rabbi, J., Masud, M., Hasan, M. K., ... & Iftee, M. A. R. (2022). Bangla natural language processing: A comprehensive analysis of classical, machine learning, and deep learning-based methods. *IEEE Access*, *10*, 38999-39044. https://doi.org/10.1109/ACCESS.2022.3165563

Suneera, C. M., & Prakash, J. (2020, December). Performance analysis of machine learning and deep learning models for text classification. In *2020 IEEE 17th India council international conference (INDICON)* (pp. 1-6). IEEE. https://doi.org/10.1109/INDICON49873.2020.9342208

Tiwari, R. S. (2024). Hate speech detection using LSTM and explanation by LIME (local interpretable model-agnostic explanations). In *Computational intelligence methods for sentiment analysis in natural language processing applications* (pp. 93-110). Morgan Kaufmann. https://doi.org/10.1016/B978-0-443-22009-8.00005-7

Yeasmin, S., Kuri, R., Rana, A. M. H., Uddin, A., Pathan, A. S. U., & Riaz, H. (2021). Multi-category Bangla news classification using machine learning classifiers and multi-layer dense neural network. *International Journal of Advanced Computer Science and Applications*, *12*(5), 88-94. https://doi.org/10.14569/IJACSA.2021.0120588

Yuan, Y., Shi, J., Zhang, Z., Chen, K., Zhang, J., Stoico, V., & Malavolta, I. (2024, April). The impact of knowledge distillation on the energy consumption and runtime efficiency of NLP models. In *Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering-Software Engineering for AI* (pp. 129-133). https://doi.org/10.1145/3644815.3644966

Zahoor, K., Bawany, N. Z., & Qamar, T. (2024). Evaluating text classification with explainable artificial intelligence. *Int J Artif Intell ISSN*, *2252*(8938), pp278-286. https://doi.org/10.11591/ijai.v13.i1.