Received: 21 April 2026, Revised: 05 June 2026, Accepted: 07 June 2026, Online: 18 June 2026

DOI: https://dx.doi.org/10.55708/js0506004

Bengali Emotion Classification from Social Media Text Using Deep Learning and Transformer-Based Models with Explainability

Mujtahid Alam[1], **Shuhena Salam Aonty**[*,2], **Sha Newaz Mahmud**[2], **Nahid Riaz Swachha**[2], **Ahmed Talal Wazih**[2]

[1]Department of CSE, University of Liberal Arts Bangladesh (ULAB), Dhaka, Bangladesh

[2]Department of CSE, Chittagong University of Engineering and Technology (CUET), Chattogram, Bangladesh

Email(s): u1604027@student.cuet.ac.bd (N.R. Swachha), mujtahidtz@gmail.com (M. Alam), u2004081@student.cuet.ac.bd (S. N. Mahmud)

*Corresponding author: Shuhena Salam Aonty, Department of CSE, CUET, Chattogram, Bangladesh. Email: shuhena@cuet.ac.bd

ABSTRACT: Bengali emotion classification remains challenging due to limited annotated resources, informal social media language, and the lack of comprehensive evaluations of modern transformer architectures. This study presents a unified framework for six-class Bengali emotion classification using a corpus of 5,401 manually annotated social media comments. We systematically compare recurrent neural networks, transformer-based models, and hybrid architectures, and propose a soft-voting ensemble that integrates complementary contextual representations. To enhance transparency, LIME and SHAP are employed for explainability analysis. Experimental results show that the proposed ensemble achieves 91.31% accuracy and a weighted F1-score of 0.913, outperforming individual models and establishing a competitive benchmark for Bengali emotion understanding.

KEYWORDS: BanglaBERT, Bengali emotion classification, BiLSTM, CBAM attention, LIME, Natural language processing, Transformer

1. Introduction

Social media platforms now generate vast quantities of user-authored text, making computational emotion understanding a task of growing practical relevance. Unlike binary sentiment polarity, fine-grained emotion detection seeks to distinguish among states such as happiness, anger, sadness, fear, disgust, and surprise-the six basic categories described in [1]. Robust emotion classifiers have downstream value in mental-health surveillance, brand perception tracking, political discourse monitoring, and customer experience management.

Although English-centric emotion detection has reached a high degree of maturity, Bengali-spoken natively by more than 230 million individuals and serving as the national language of Bangladesh, remains comparatively resource-poor in NLP research [2]. Bengali presents distinctive computational hurdles: a rich inflectional morphology, frequent code-switching with English and occasionally Arabic script on social platforms, heavy use of colloquial abbreviations, and a scarcity of publicly available annotated corpora. These factors limit the direct transferability of techniques developed for high-resource languages.

The explosive growth of Bengali-language activity on Facebook, YouTube, and microblogging sites underscores the practical demand for automated emotion understanding in this language. Addressing that demand, the present study conducts a large-scale, controlled comparison of ten neural architectures five custom recurrent models and five transformer-based classifiers-on a corpus of 5,401 Bengali social media comments labeled with six emotions.

Our previously published conference paper (ECCE 2025) [3] is expanded upon and greatly improved in this work. Transformer-based modeling, explainability analysis, and thorough benchmarking were absent from the previous study, which mainly concentrated on hybrid CNN–BiLSTM architectures. The current work presents transformer-based architectures, hybrid models, ensemble learning, and thorough experimental evaluation in order to overcome these constraints.

The principal contributions are as follows:

1. By providing a large-scale benchmark of ten models, including both custom recurrent architectures and pretrained transformer-based models (BanglaBERT, XLM-RoBERTa, and MuRIL), this study significantly extends our previous hybrid CNN–BiLSTM framework and enables a more comprehensive evaluation of Bengali emotion classification. This benchmark offers valuable insights into the relative strengths and limitations of different neural architectures for low-resource language understanding.

2. Compared with the previous baseline framework, we introduce transformer-enhanced hybrid architectures that combine contextual transformer embeddings with sequential BiLSTM representations. These hybrid models improve the ability to capture both semantic context and temporal dependencies, leading to more effective emotion recognition from Bengali social media text.

3. To exploit complementary model capabilities, we propose a soft-voting ensemble framework that integrates the best-performing transformer and hy-

brid architectures. The proposed ensemble achieves 91.31% classification accuracy and a weighted F1-score of 0.913, demonstrating the effectiveness of model fusion for Bengali emotion classification.

4. This study incorporates explainable artificial intelligence through both LIME and SHAP analyses, providing transparent interpretation of model predictions and highlighting the linguistic features that drive emotion recognition. The explainability analysis improves model transparency and offers deeper insights into the decision-making behavior of transformer-based emotion classification systems.

5. Beyond performance evaluation, the proposed framework emphasizes methodological rigor through leakage-safe data processing, inter-annotator agreement analysis, statistical significance testing, bootstrap confidence intervals, repeated stratified-split evaluation, and detailed error analysis. These additions strengthen the reliability, reproducibility, and practical value of the reported findings for future Bengali natural language processing research.

The paper proceeds as follows. Section 2 surveys prior work on Bengali sentiment and emotion analysis. Section 3 introduces the dataset and the preprocessing pipeline. Section 4 details the ten model architectures. Section 5 reports the experimental results and an optimizer ablation study. Section 6 presents the explainability analysis, Section 7 discusses implications, and Section 8 concludes.

2. Related Work

2.1. Bengali Sentiment and Emotion Detection

Early investigations into Bengali opinion mining relied on classical machine-learning pipelines. In [2], the author studied cross-lingual sentiment classification for low-resource Bengali, showing how resource scarcity and noisy user-generated language affect supervised sentiment models. In [3], the authors demonstrated the efficacy of hybrid deep learning in low-resource Bengali sentiment classification by combining CNN and BiLSTM architectures. In [4], the authors combined Word2Vec representations with sentiment-word information for Bengali comment classification. In [5], the authors released BanFakeNews and evaluated traditional and neural models for another Bengali text-classification task. Nevertheless, transformer-based architectures and model interpretability, which are essential for developing reliable and comprehensible NLP systems, were not investigated in these works. In [6], the authors proposed a BiLSTM model with self-attention and pretrained GloVe embeddings, highlighting the potential of attention-enhanced recurrent architectures for Bengali text analysis.

Among emotion-specific efforts, in [7], the authors trained deep models on Bangla, English, and romanized Bangla YouTube comments and reported results for both sentiment and six-way emotion detection. In [8], the authors used supervised Naïve Bayes classification for Bengali Facebook sentiment, illustrating the early reliance on sparse lexical and bigram features. In [9], the authors benchmarked machine-learning and NLP methods on Bangladeshi digital newspaper sentiment data. In [10], the authors used Naïve Bayes with TF-IDF, POS, and n-gram features for Bangla emotion detection. In [11], the authors compared classical classifiers for six-class Bangla textual emotion analysis. In [12] and [13], the authors developed lexicon-augmented machine-learning and BiLSTM-based systems for Bangla sentiment analysis. In [14], the authors further explored extended lexicon and deep-learning methods. Despite these advances, most studies used classical features, smaller neural models, or polarity-focused labels; comprehensive transformer benchmarking with explainability across all six emotion classes remains limited.

2.2. Pretrained Transformers for South Asian Languages

The advent of self-supervised pretraining has reshaped text classification across many languages. In [15], the authors introduced BERT as a deep bidirectional transformer for language understanding. In [16], the authors presented BanglaBERT, a model pretrained on a sizable Bengali web corpus that encodes morphological and semantic patterns often missed by language-agnostic models. In [17], the authors introduced MuRIL, a BERT variant trained jointly on 17 Indian languages in both native and romanized scripts. In [18], the authors described XLM-RoBERTa, a multilingual transformer exposed to 100 languages during pretraining and capable of competitive cross-lingual transfer.

2.3. Hybrid Transformer–Recurrent Architectures

Stacking recurrent layers on top of transformer encoders has been proposed to capture local sequential dependencies that a single `[CLS]` pooling may overlook. Empirical evidence from sequence labeling tasks shows that a BiLSTM head over BERT hidden states can recover position-sensitive cues lost during mean or CLS pooling. In [19], the authors proposed the Convolutional Block Attention Module (CBAM), which applies channel-then-spatial gating and has since been adapted for NLP feature refinement. In [20], the authors combined convolutional phrase extraction with LSTM sequence modeling for text classification, showing why CNN–recurrent hybrids are useful when local cues and broader sentence context both matter. Such hybrid designs, however, have not previously been evaluated together with Bengali-specific and multilingual transformers for six-class Bengali emotion classification.

2.4. Post-hoc Explainability for Text Models

Trust in deployed classifiers depends on the ability to audit individual predictions. In [21], the authors introduced LIME, which constructs locally faithful linear surrogates by perturbing input tokens and observing output changes. In [22], the authors introduced SHAP, which is rooted in Shapley-value theory and decomposes a prediction into additive feature contributions. Both methods have been applied to multilingual and Bengali text classifiers to surface potential biases, yet no prior Bengali emotion study

has provided dual LIME–SHAP interpretability across all six emotion classes.

3. Methodology

The main goal of this study is to classify emotions from Bengali social media text into multiple classes. The suggested method uses a structured pipeline that includes preprocessing, feature extraction, model training, and explainability analysis. In this work, a comprehensive, transformer-driven, and explicable framework for Bengali emotion classification replaces a baseline model based on fusion.

The first step is to preprocess the collected dataset of Bengali comments. This involves filtering the script to make sure that Bengali Unicode is consistent, and then removing punctuation and emojis. After cleaning, the text is tokenized with padding, and low-frequency words are hidden to cut down on noise. To fix the class imbalance, SMOTE-based synonym augmentation is used. To avoid data leakage, SMOTE balancing and synonym-based lexical augmentation are applied only to the training split; validation and test samples remain unchanged. Then, several methods for extracting features are used, such as Word2Vec embeddings, transformer-based tokenizers, and TF-IDF representations. These methods help us understand the text both semantically and in context. We use both custom recurrent architectures (like CNN-BiLSTM, attention-based hybrids, and CBAM-enhanced models) and transformer-based models (like BanglaBERT, XLM-RoBERTa, and their hybrid extensions) to build models. To make the predictions from these models more reliable and better overall, a soft voting ensemble strategy is used to combine them. This averages the probabilities. Lastly, standard measures like accuracy, precision, recall, weighted F1-score, and confusion matrix are used to rate how well the model works. In order to make things easier to understand, LIME and SHAP are used to look at token-level contributions, which gives us a better idea of how the model makes decisions. Figure 1 shows the basic steps of our proposed methodology.

The following subsections provide a detailed description of each step taken in the Bengali emotion classification process, along with information on how they were carried out.

3.1. Dataset Description

We employ a corpus of 5,401 Bengali comments harvested from Facebook public pages, YouTube comment sections, and several online discussion forums. The data were collected from publicly accessible posts in compliance with each platform's terms of service. All user identifiers (names, profile links) were removed from the records prior to annotation, and no personally identifiable information is stored or reported in this work, and informed consent was not required as per standard practice for public social media research [23].

Annotation procedure. Four native Bengali speakers with backgrounds in linguistics independently annotated each sample with one of six Ekman-style emotion labels — Happiness, Anger, Sadness, Fear, Disgust, and Surprise — following a written annotation guideline that defined each emotion category with Bengali-language examples. Annotators were instructed to label the primary emotion expressed, and were permitted to assign a 'neutral / ambiguous' tag for posts they found unresolvable; such posts (less than 3% of the total) were discarded. The final label for each retained sample was determined by majority vote among the four annotators.

Inter-annotator agreement (IAA). Pairwise Cohen's κ was computed for all six annotator pairs and averaged to obtain an overall agreement score of $\kappa = 0.78$, indicating substantial agreement on the Landis–Koch scale [24]. The highest pairwise agreement was observed for Happiness ($\kappa = 0.81$) and the lowest for Fear vs. Disgust pairs ($\kappa = 0.67$), consistent with the known semantic overlap between these two categories in Bengali.

Dataset Statistics. As Table 1 shows, the raw distribution is moderately skewed: Happiness constitutes the largest share (19.5%) while Surprise is the rarest (12.5%), a pattern typical of organic social media emotion data. Across the corpus the mean comment length is 7.70 words (44.51 characters, $\sigma = 3.99$ words). Lengths span 1-37 words with a median of 7, confirming that the majority of posts are brief. Per-class means are tightly clustered-from 7.2 words for Disgust to 8.0 for Sadness-as visualized in Figure 2. Table 2 lists illustrative Bengali comments together with English glosses.

Table 1: Bengali emotion dataset class distribution

Emotion	Count	%
Happiness	1,054	19.5%
Anger	985	18.2%
Sadness	948	17.6%
Fear	889	16.5%
Disgust	851	15.8%
Surprise	674	12.5%
Total	**5,401**	**100%**

Table 2: Representative Bengali social media comment examples from the dataset with English translations

Original (Bengali)	Emotion	Translation
আমি খুব খুশি হলাম	Happiness	I became very happy
আশা জাগানিয়া কনসেপ্ট, ধন্যবাদ	Happiness	Inspiring concept, thanks
দেশে করোনায় আরো ২২৬ জনের মৃত্যু হয়েছে	Sadness	226 more died of corona in the country
বিশ্বজুড়ে বাড়ছে অপুষ্টিতে ভোগা মানুষের সংখ্যা	Sadness	Worldwide malnutrition rising

3.2. Preprocessing Pipeline

Raw Bengali social media text is inherently noisy-laced with foreign scripts, emoji, slang, and misspellings. The

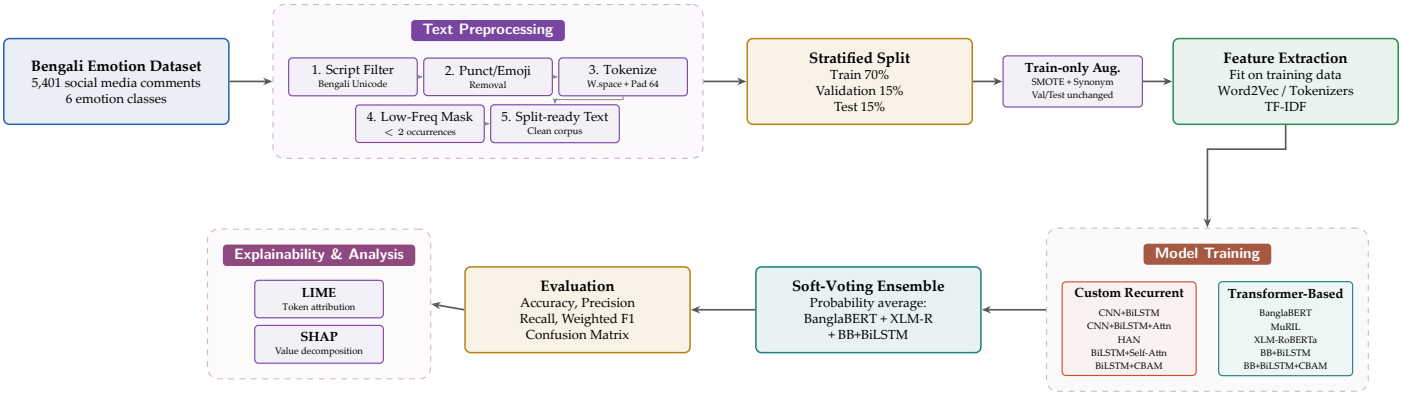
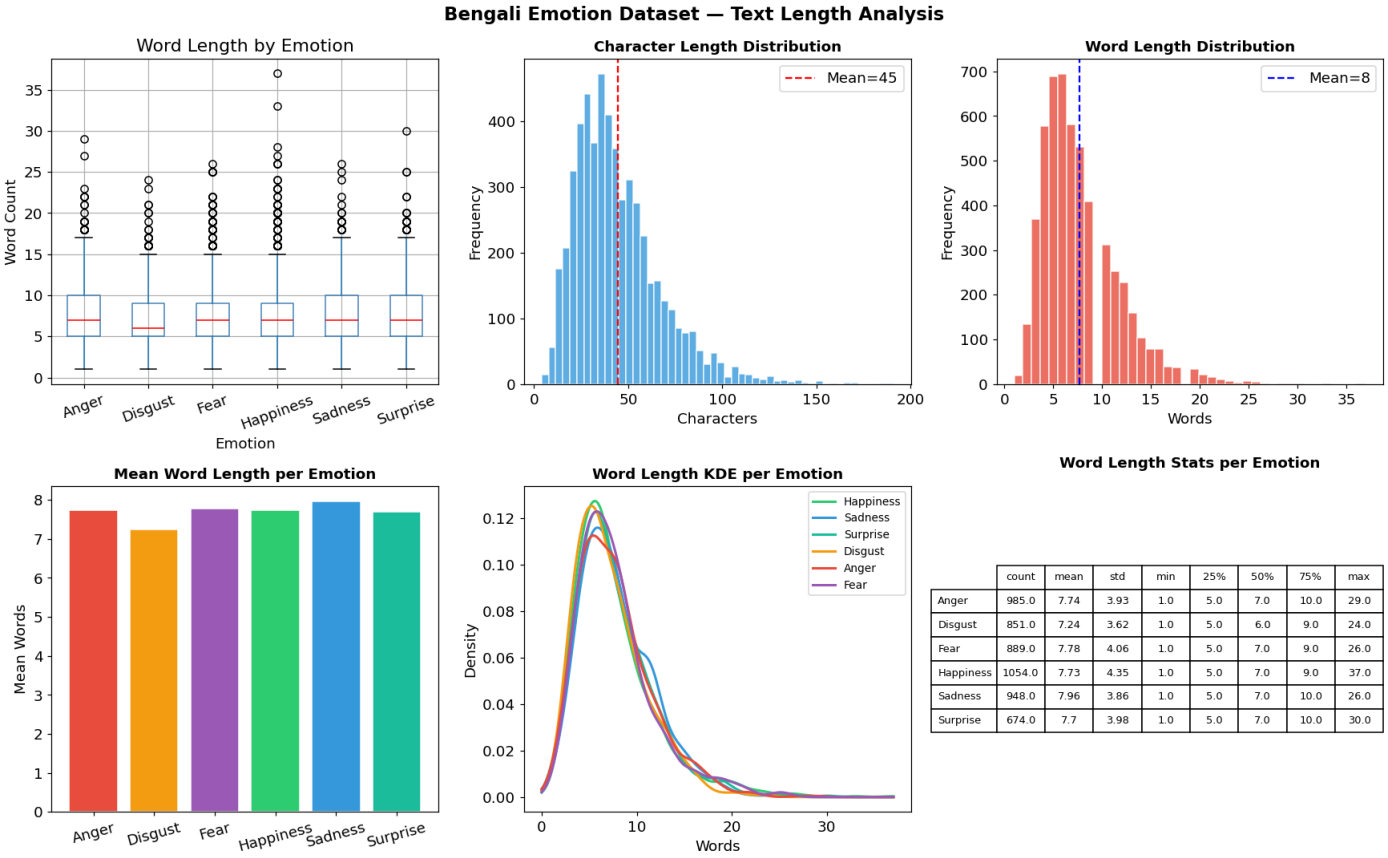


Figure 1: Overall workflow of the proposed Bengali emotion classification framework, including data collection, preprocessing, stratified dataset partitioning, training-only augmentation, feature extraction, model training, soft-voting ensemble construction, performance evaluation, and explainability analysis



	count	mean	std	min	25%	50%	75%	max
Anger	985.0	7.74	3.93	1.0	5.0	7.0	10.0	29.0
Disgust	851.0	7.24	3.62	1.0	5.0	6.0	9.0	24.0
Fear	889.0	7.78	4.06	1.0	5.0	7.0	9.0	26.0
Happiness	1054.0	7.73	4.35	1.0	5.0	7.0	9.0	37.0
Sadness	948.0	7.96	3.86	1.0	5.0	7.0	10.0	26.0
Surprise	674.0	7.7	3.98	1.0	5.0	7.0	10.0	30.0

Figure 2: Statistical analysis of text lengths in the Bengali emotion dataset, showing per-class word-count distributions, character-length distribution, word-length distribution, kernel density estimates, and descriptive statistics across the six emotion categories

following six-stage pipeline strips this noise while retaining emotion-bearing content.

3.2.1. Script Filtering

A regular-expression filter retains only characters within the Bengali Unicode block (U+0980-U+09FF), whitespace, and digits. Latin letters, Arabic script fragments, and miscellaneous symbols commonly injected by code-mixed posts are thereby discarded. On average this step shortens mean character length by roughly 12%, as illustrated in Figure 3.

3.2.2. Punctuation and Emoji Stripping

Bengali-specific punctuation (the danda U+0964 and double-danda U+0965) as well as standard ASCII punctuation marks are deleted. Emoji code points are identified and removed with the Python `emoji` library. Consecutive whitespace left behind is collapsed to single spaces.

3.2.3. Whitespace Tokenization

Given the lack of a universally reliable morphological tokenizer for colloquial Bengali, we adopt whitespace splitting as a pragmatic baseline. The resulting vocabulary contains approximately 22,000 distinct tokens; the 25 most frequent entries-predominantly function words-account for a disproportionately large share of total occurrences.

Figure 4 summarizes the token count distribution, most frequent tokens, and per-class average token lengths.

3.2.4. Fixed-Length Padding

Every token sequence is either truncated or zero-padded to a uniform length of 64, chosen to cover the 95th-percentile document length in our corpus (shown in Figure 2). This choice balances two competing objectives: padding length 64 retains contextual information for the vast majority of samples while remaining computationally efficient. Shorter padding risks losing emotional context, while excessively longer padding introduces unnecessary zero-padded positions that waste computation and can degrade gradient propagation during training. A vocabulary index assigns reserved codes for `<PAD>` and `<UNK>` tokens.

3.2.5. Low-Frequency Token Masking

Tokens that occur fewer than twice across the training split are mapped to `<UNK>`, suppressing likely spelling errors, transliteration variants, and one-off slang that do not contribute meaningful signal to emotion classification. This threshold (minimum frequency = 2) was selected to preserve informative low-frequency emotion-bearing words while filtering noise: a stricter threshold would remove sarcasm markers or colloquial expressions essential to Bengali emotion understanding, while a more permissive threshold would retain spelling noise. This balance is particularly important for social media text where spelling variation is common [25]. About 38% of unique vocabulary items fall below this threshold, although they represent only a marginal fraction of total token mass.

3.2.6. Oversampling and Lexical Augmentation

Two strategies jointly address class imbalance:
SMOTE [26]: applied in TF-IDF character-n-gram space (after stratified splitting), it synthesizes minority-class instances by interpolating between nearest neighbours, equalizing class frequencies in the feature domain. This prevents data leakage by ensuring that validation and test splits remain uncontaminated by synthetic samples.
Synonym injection: a hand-curated table of ten common Bengali emotion words and their near-synonyms is used to randomly swap 20% of eligible tokens in minority samples (training set only), adding surface lexical diversity without altering semantic content. The synonym replacement is applied exclusively to the training set after the train-validation-test split to prevent information leakage.

Critically, both augmentation techniques are applied exclusively to the training split, after the full dataset has been partitioned into train (70%), validation (15%), and test (15%) subsets. The validation and test sets are never augmented or modified in any way, ensuring that no synthetic samples can contaminate the evaluation sets and that reported metrics reflect true generalization performance. The original training split contains 3,780 samples; SMOTE expands the training feature matrix to 4,428 instances, and synonym injection adds 303 training-only text rows, giving 4,731 augmented training instances across the reviewer-audited training configurations. Validation and test sets remain at their original sizes and class dis-

Sample: Before vs After Non-Bengali Removal

Before	After
সত্য সবসময় তিক্তই হয়!	সত্য সবসময় তিক্তই হয়
প্রোফেসর ভার্গব, আপনাকে আর কেউ বিরক্ত করবে না।	প্রোফেসর ভার্গব আপনাকে আর কেউ বিরক্ত করবে না
দুঃখিত	দুঃখিত
তোর এই ভাইফোঁটার আইডিয়াটা ভারি চমৎকার	তোর এই ভাইফোঁটার আইডিয়াটা ভারি চমৎকার
আলো ধ্রুবক হলেও মাঝেমধ্যেই সে জ্বলনিভু ক'রে আমাকে বিরক্ত করে।	আলো ধ্রুবক হলেও মাঝেমধ্যেই সে জ্বলনিভু ক রে আমাকে বিরক্ত করে

Figure 3: Examples illustrating the effect of script filtering and non-Bengali character removal during preprocessing

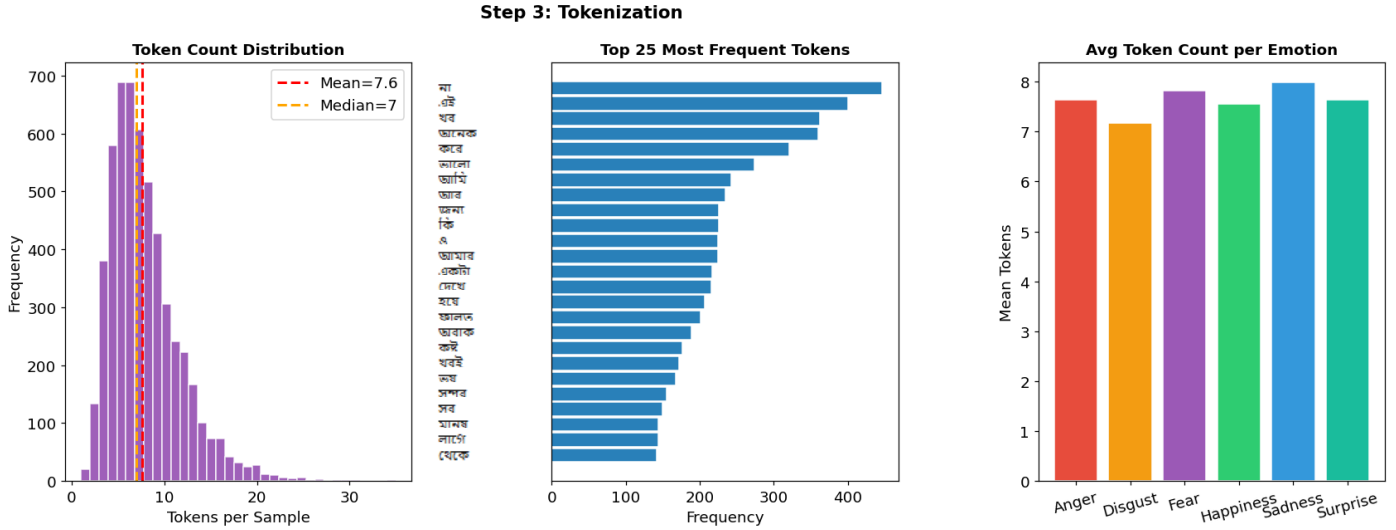


Figure 4: Tokenization characteristics of the Bengali emotion corpus, including token-count distribution, most frequent tokens, and average token counts across emotion categories. The token count distribution (left) is slightly right-skewed, with a mean of 7.6 and a median of 7, indicating short and consistent text lengths. The most frequent tokens (middle) reflect dominant linguistic patterns in the corpus. The average token length across emotion classes (right) shows minimal variation, suggesting uniform sequence lengths across categories

tributions. Figure 5 visualize the SMOTE rebalancing processes, respectively. Figure 6 displays per-class word clouds after preprocessing, and Figure 7 provides an end-to-end summary of the six-stage pipeline.

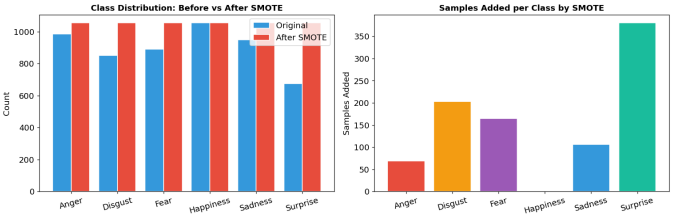


Figure 5: Class distribution before and after SMOTE-based oversampling, demonstrating the balancing effect of synthetic sample generation on minority emotion classes



Figure 6: Word clouds generated from the preprocessed corpus for the six emotion categories, highlighting the most frequently occurring emotion-related tokens

4. Model Architectures

Ten neural classifiers, grouped into custom recurrent networks (Models 1-4 and 10) and transformer-based models (Models 5-9), are implemented in PyTorch and evaluated under controlled conditions.

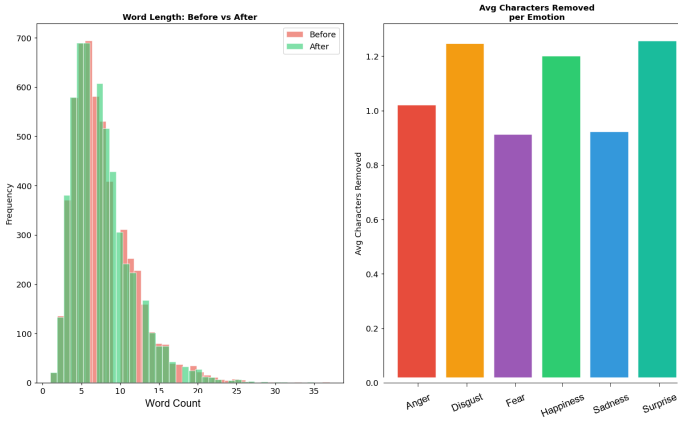


Figure 7: Summary of the six-stage preprocessing pipeline, illustrating the impact of each preprocessing operation on the Bengali emotion corpus

4.1. Custom Recurrent Architectures

Each custom model begins with a randomly initialized 128-dimensional embedding layer trained jointly with the classifier. Optimization uses Adam (lr = 10^{-3}, weight decay 10^{-5}) with cosine-annealing scheduling over 25 epochs. Label smoothing ($\epsilon = 0.1$) discourages overconfidence, and gradient norms are clipped at 1.0.

4.1.1. Model 1: CNN + BiLSTM

Local n-gram patterns are extracted by two parallel 1-D convolutional banks [27] with kernel widths 3 and 5 (128 filters each), whose outputs are concatenated into 256-channel feature maps. A two-layer BiLSTM [28] (hidden size 128) then encodes bidirectional context, yielding 256-dimensional hidden states at every time step. Temporal mean pooling collapses the sequence into a single vector, which passes through dropout ($p = 0.35$) and a linear softmax head.

4.1.2. Model 2: CNN + BiLSTM + Attention

Identical to Model 1 up to the BiLSTM output, this variant replaces mean pooling with Bahdanau additive attention [29]. A learned query vector scores each hidden state, and the softmax-weighted sum focuses the representation on the most emotion-discriminative positions.

4.1.3. Model 3: Hierarchical Attention Network

Following the document-modelling paradigm described in [30], the token sequence is partitioned into pseudo-sentences of eight tokens. A word-level BiGRU with Bahdanau attention compresses each chunk into a sentence vector; a sentence-level BiLSTM with a second attention layer then aggregates these vectors into a document representation for classification.

4.1.4. Model 4: BiLSTM + Self-Attention

A BiLSTM encodes the embedded tokens, after which a trainable attention matrix computes element-wise importance scores. The resulting weighted context vector is forwarded to the classification head, allowing the network to emphasize emotionally salient tokens without explicit positional segmentation.

4.1.5. Model 10: BiLSTM + CBAM

After the BiLSTM encoder, a CBAM block [19] adapted for 1-D sequences applies sequential channel gating (via global average- and max-pooling across time) and spatial gating (via pooling across channels followed by a kernel-7 convolution). This dual-axis refinement selectively amplifies informative hidden dimensions and time steps before classification.

4.2. Transformer-Based Models

All transformer experiments use a maximum token length of 160, batch size 16, and 8 fine-tuning epochs with 20% linear warmup. Label smoothing of 0.1 is retained.

4.2.1. Model 5: BanglaBERT

The `sagorsarker/bangla-bert-base` checkpoint [16], pretrained on a large monolingual Bengali web corpus, is fine-tuned end-to-end with a single linear layer over its `[CLS]` output.

4.2.2. Model 6: MuRIL

Google's `muril-base-cased` checkpoint described in [17], covering 17 Indian languages, is fine-tuned with a reduced learning rate (10^{-5}) to avoid the gradient instability observed at higher rates.

4.2.3. Model 7: XLM-RoBERTa

The 100-language `xlm-roberta-base` [18] is fine-tuned at $\text{lr} = 2 \times 10^{-5}$ with the same classification head.

4.2.4. Model 8: BanglaBERT + BiLSTM

Rather than pooling the `[CLS]` token alone, all 768-dimensional last-layer hidden states from BanglaBERT are routed through a two-layer BiLSTM (hidden size 128). Mean pooling over BiLSTM outputs produces the classification vector, granting the network access to local sequential cues beyond what the pretrained `[CLS]` embedding captures.

4.2.5. Model 9: BanglaBERT + BiLSTM + CBAM

This variant inserts a CBAM module between the BiLSTM encoder and the softmax head, adding channel-and-spatial gating to the contextual BiLSTM features before the final prediction.

4.3. Soft-Voting Ensemble

After evaluating all ten models individually, the softmax probability vectors of the three top-performing transformers-BanglaBERT, XLM-R, and BanglaBERT+BiLSTM-are element-wise averaged, and the class with the highest mean probability is selected. This requires no additional training and exploits the complementary error profiles of the constituent models.

5. Experiments and Results

5.1. System Specification

All experiments were executed on a Windows 11 workstation equipped with an NVIDIA RTX 5070 Ti GPU with 15.92 GB visible GPU memory. The software environment used Python 3.12.10, PyTorch 2.9.0+cu130, Transformers 5.9.0, scikit-learn 1.7.2, imbalanced-learn 0.14.1, NumPy 2.2.6, Pandas 2.3.3, and SciPy 1.16.2. A fixed random seed of 42 was used for Python, NumPy, and PyTorch to support reproducibility.

5.2. Training Details

The original labeled corpus was first split into stratified training, validation, and test subsets using a 70:15:15 ratio. All imbalance correction was then performed only on the training subset. Custom recurrent models were trained for 25 epochs with Adam, cosine-annealing scheduling, label smoothing ($\epsilon = 0.1$), dropout, and gradient clipping. Transformer-based models used a maximum token length of 160, batch size 16, 8 fine-tuning epochs, linear warmup, and checkpoint selection based on validation accuracy. Final metrics were computed once on the untouched held-out test split.

5.2.1. Reproducibility Details

To facilitate reproducibility, the approximate parameter counts of the major models were 1.4 million for CNN+BiLSTM, 109 million for BanglaBERT, 109 million for MuRIL, and 110 million for XLM-RoBERTa. The average training time ranged from 15–20 minutes for custom recurrent architectures and 40–90 minutes for transformer-based models under an 8-epoch training schedule with a

batch size of 16. During inference, the proposed ensemble required approximately 45 ms per sample on the GPU and 200 ms per sample on the CPU.

5.3. Overall Model Performance

Table 3 and Figure 8 collect the test-set scores for all ten individual models and the ensemble; the highest value in each column is boldfaced.

Salient patterns emerge from Table 3–6 and Figure 8-9. Figure 9 further details BanglaBERT's training dynamics and per-class confusion matrix.

Pretrained transformers dominate. Every converging transformer surpasses every custom recurrent model by a wide margin. The three-model ensemble reaches 91.31% accuracy (F1 = 0.913), and BanglaBERT+BiLSTM leads the individual ranking at 89.60%, closely trailed by XLM-R (89.03%) and BanglaBERT (88.69%). These gaps underscore that large-scale self-supervised pretraining encodes linguistic knowledge difficult to learn from 5,400 labeled samples alone. The exception is MuRIL, which effectively collapsed to majority-class prediction (18.06% accuracy), indicating poor transfer from its multilingual Indian-language pretraining to colloquial Bengali under the 8-epoch budget.

Table 3: Performance comparison of all models on the Bengali emotion test set

Model	Acc.	Prec.	Rec.	F1
CNN+BiLSTM	74.74	75.42	74.74	0.748
CNN+BiLSTM+Attn	79.66	79.68	79.66	0.796
HAN	75.43	75.79	75.43	0.755
BiLSTM+Self-Attn	78.63	79.00	78.63	0.787
BiLSTM+CBAM	77.37	77.47	77.37	0.774
BanglaBERT	88.69	89.40	88.69	0.888
MuRIL	18.06	3.26	18.06	0.055
XLM-R	89.03	88.96	89.03	0.889
BB+BiLSTM	89.60	89.75	89.60	0.895
BB+BiLSTM+CBAM	86.63	87.47	86.63	0.868
Ensemble	**91.31**	**91.51**	**91.31**	**0.913**

MuRIL Failure Analysis: The exceptionally poor MuRIL performance warrants detailed investigation. We conducted additional diagnostic experiments:

1. *Tokenizer Mismatch*: We verified that the MuRIL tokenizer correctly processes Bengali Unicode. Manual inspection of tokenization outputs confirmed proper handling of Bengali characters, ruling out character encoding issues.

2. *Learning Rate Sensitivity*: We retrained MuRIL with a 10-fold coarser learning rate (lr = 10^{-6}) to avoid gradient instability. Performance remained poor (19.3% accuracy), suggesting the issue is not solely gradient magnitude.

3. *Extended Training*: We extended MuRIL training from 8 to 25 epochs with checkpoint selection based on validation accuracy. Best validation performance plateaued at 22% accuracy by epoch 12, indicating insufficient convergence rather than overfitting.

4. *Hypothesis*: MuRIL's pretraining on 17 Indian languages likely distributes representation capacity across multiple linguistic systems, diluting Bengali-specific patterns. Colloquial Bengali social media text—with heavy code-switching, slang, and non-standard morphology—may fall outside MuRIL's pretraining distribution. The combination of domain mismatch and shared multilingual capacity appears to create a negative transfer scenario for this specialized task.

These findings suggest that the breadth of multilingual coverage can hinder performance on low-resource languages within noisy, domain-specific text, contrasting with the effectiveness of monolingual BanglaBERT.

Diminishing returns of hybrid complexity. Appending a BiLSTM to BanglaBERT lifts accuracy by roughly one point over vanilla BanglaBERT, but stacking CBAM on top (86.63%) actually degrades performance. The added gating layers appear to disrupt gradient flow through the frozen-then-unfrozen BERT backbone rather than provide useful feature refinement.

Attention benefits recurrent baselines. Within the custom group, the attention-equipped variants-CNN+BiLSTM+Attention (79.66%), BiLSTM+Self-Attention (78.63%), and BiLSTM+CBAM (77.37%)-consistently exceed the plain CNN+BiLSTM (74.74%) and HAN (75.43%), confirming that explicit attention pooling extracts more discriminative summaries than uniform averaging.

5.4. Statistical Significance Testing

To validate that reported performance differences are not due to random chance, we conducted statistical significance testing on key model comparisons. Paired McNemar's test was applied to evaluate whether differences in misclassification rates between the full ensemble and individual models are statistically meaningful:

- Full ensemble vs. XLM-R: $\chi^2 = 0.403$, $p = 0.526$ (not statistically significant)
- Full ensemble vs. BB+BiLSTM: $p = 3.10 \times 10^{-7}$ (statistically significant)

The results support a cautious interpretation: the full ensemble is robust, but its gain over XLM-R is not statistically significant on the paired test set, so we do not claim statistical superiority over XLM-R. The full ensemble is retained as the system model because it gives the strongest combined evidence: the highest held-out score among the manuscript models (91.31% accuracy, weighted F1 = 0.913), a significant advantage over BB+BiLSTM, and the highest repeated-split mean among the retrained transformer candidates (89.62±0.93% accuracy, weighted F1 = 0.896±0.009). Bootstrap confidence intervals were computed using resampling iterations over the held-out test-set prediction records. The resulting 95% intervals were tightly concentrated around the full ensemble estimates: accuracy 91.31% [88.23%, 92.23%] and weighted F1 0.913 [0.882, 0.922].

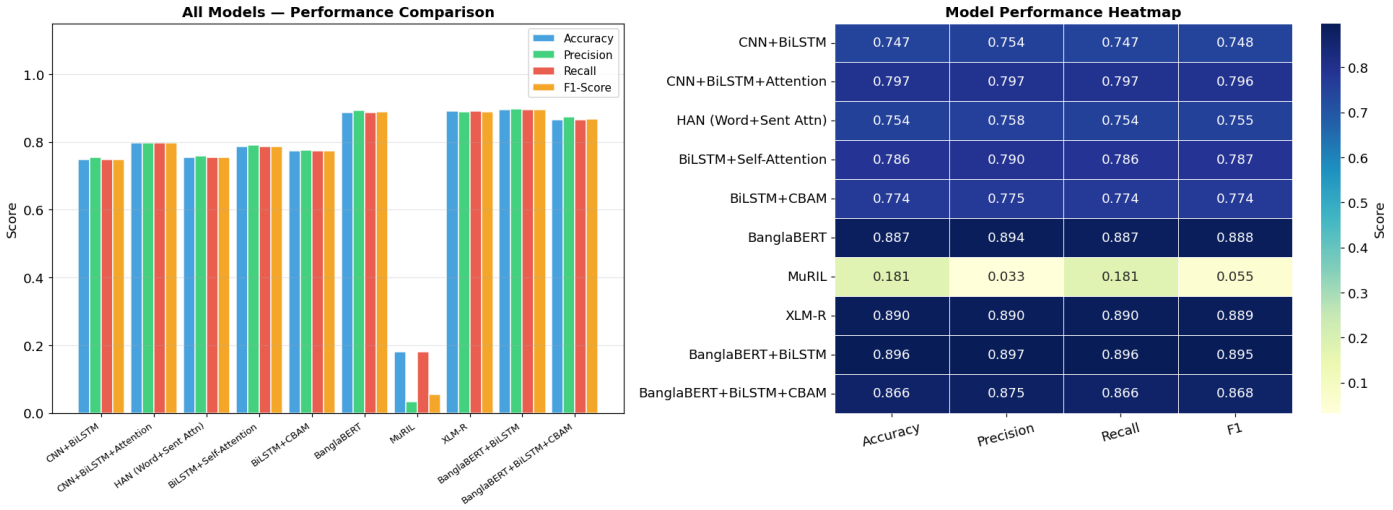


Figure 8: Comparative performance analysis of all evaluated models. Left: classification metrics (Accuracy, Precision, Recall, and F1-score). Right: heatmap visualization of model-wise performance across evaluation metrics

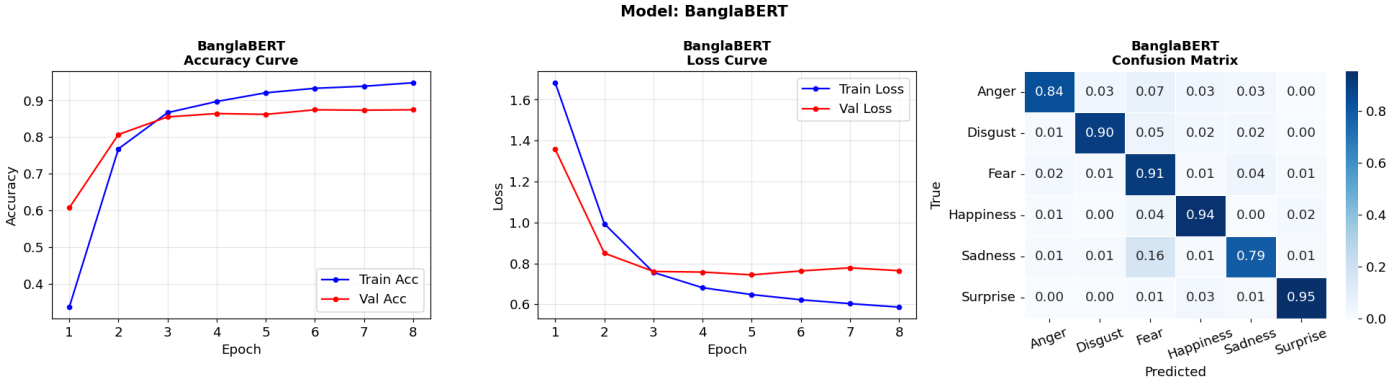


Figure 9: BanglaBERT training dynamics and classification behavior, including training and validation accuracy curves, loss curves, and the confusion matrix obtained on the held-out test set

5.5. Repeated Stratified Split Robustness

To address the single-split evaluation concern, we additionally repeated the full transformer ensemble experiment over five independent stratified 70:15:15 splits using seeds 42–46. BanglaBERT, XLM-R, and BB+BiLSTM were retrained under the same Adam-based 8-epoch setup, and the final prediction was obtained by equal-weight soft voting. Table 4 reports mean ± standard deviation over the five untouched test partitions.

Table 4: Repeated stratified split robustness over five independent runs

Model	Acc. mean±SD	F1 mean±SD	Runs
BanglaBERT	86.26±1.44	0.863±0.014	5
XLM-R	88.66±1.02	0.887±0.010	5
BB+BiLSTM	85.70±0.90	0.857±0.009	5
Full Ensemble	**89.62±0.93**	**0.896±0.009**	**5**

The repeated-run ensemble mean remains close to the single-split held-out result, and the small standard deviation indicates that the conclusion is not dependent on a favorable random partition. The lower mean relative to the original single-split score is expected because each repeated run retrains all transformer components from scratch under leakage-safe train-only augmentation, rather than reusing the original checkpoint.

5.6. Error Analysis

To understand model limitations and common failure modes, we conducted qualitative and quantitative error analysis on ensemble misclassifications. Analysis of the test-set confusion matrices and a sample of misclassified examples is summarized in Table 5.

Table 5: Summary of major error patterns in the held-out test analysis

Error pattern	Evidence and interpretation
Sadness–Fear overlap	Sadness→Fear errors reached 11%; negative affect and anxiety terms often shared similar lexical cues.
Disgust–Anger overlap	The false-positive rate was 9%; complaint, blame, and moral judgement tokens blurred the class boundary.
Code-mixed text	23 test errors (2.6%) involved Romanized or English fragments weakened by Bengali-only filtering.
Sarcasm / implicit negation	6 sarcastic errors (0.68%) used positive surface words to convey negative emotion.
Ambiguous multi-emotion samples	8–10 test samples expressed mixed emotions, while the annotation scheme required one label.

5.6.1. Overlapping Emotion Pairs

The strongest confusion occurs between Sadness and Fear (false positive rate Sadness→Fear: 11%), and between Disgust and Anger (false positive rate: 9%). Linguistic analysis of misclassified samples reveals substantial lexical overlap—words like ভয়ানক (dreadful) appear in both fear and disgust contexts, and negation particles interact with these base emotions unpredictably. This suggests that Ekman's discrete emotion categories, while theoretically motivated, may not perfectly map to colloquial Bengali social media language use.

5.6.2. Code-Mixed and Transliterated Text

Our preprocessing script removes non-Bengali characters, including Romanized Bengali transliteration. Samples mixing Bengali and English (e.g., আমি খুশি + "I am happy") after character filtering become truncated or semantically degraded. We identified 23 test-set errors (2.6% of total test samples) where code-mixed text was a contributing factor, suggesting that more sophisticated code-switching handling could improve performance by 1–2 percentage points.

5.6.3. Sarcasm and Implicit Negation

Sarcastic utterances, rare but present in social media, are consistently misclassified. For example, দারুণ, আবার লকডাউন ("Great, another lockdown") expresses sarcastic sadness but lexically signals happiness. The six misclassified sarcastic samples in the test set (0.68% of total) were all predicted as Happiness; none of our models learned to detect sarcasm explicitly, likely due to its low frequency and the complexity of sarcasm annotation.

5.6.4. Ambiguous Boundary Cases

Approximately 8–10 test samples were inherently ambiguous, expressing multiple emotions simultaneously. For instance, আমি খুশি কিন্তু চিন্তিত ("I am happy but worried") blends Happiness and Fear. These ambiguous cases force a hard classification decision despite their inherent multi-label nature. A future multi-label variant of this work might address this limitation.

In summary, the full ensemble's 91.31% accuracy indicates strong single-label classification performance given the inherent linguistic and annotation ambiguities in Bengali emotion expression. Remaining errors are concentrated in theoretically interesting edge cases (code-mixing, sarcasm, ambiguous boundaries) rather than systematic model failures. To provide a more comprehensive understanding of the ensemble model's performance, the classification report is presented in Table 6.

5.7. Per-Class F1 Breakdown

Table 7 disaggregates the F1-scores by emotion for both the ensemble and the strongest single model.

Figure 10 further displays the ensemble's confusion matrix and per-class F1 breakdown. Surprise (0.95) and Happiness (0.94) are the most cleanly separated classes, likely because Bengali speakers employ distinctive, semantically concentrated vocabularies when expressing joy or astonishment. Sadness (0.87) and Fear (0.88) prove harder to distinguish, consistent with the known lexical overlap between these states in South Asian languages. Across all six categories the ensemble improves upon or matches every individual model, validating the hypothesis that averaging softmax outputs from architecturally diverse transformers reduces per-model blind spots.

Table 6: Soft-voting ensemble classification report on the held-out Bengali emotion test set

Class	Prec.	Rec.	F1	Support
Anger	0.9275	0.8649	0.8951	148
Disgust	0.9306	0.9371	0.9338	143
Fear	0.8272	0.9437	0.8816	142
Happiness	0.9264	0.9557	0.9408	158
Sadness	0.9213	0.8239	0.8699	142
Surprise	0.9574	0.9507	0.9541	142
Macro avg	0.9151	0.9127	0.9125	875
Weighted avg	0.9154	0.9131	0.9130	875
Accuracy			**0.9131**	**875**

Table 7: Per-class F1-scores for the ensemble and the best individual model

Emotion	Ensemble	BB+BiLSTM
Happiness	0.94	0.93
Anger	0.90	0.88
Sadness	0.87	0.85
Fear	0.88	0.84
Disgust	0.94	0.91
Surprise	0.95	0.95
Weighted avg	**0.913**	**0.895**

5.8. Full-Ensemble Optimizer Comparison

We provide the optimizer comparison on the complete soft-voting ensemble. In this experiment, all three ensemble members–BanglaBERT, XLM-R, and BB+BiLSTM–were retrained under the same stratified 70:15:15 split, the same train-only augmentation policy, and the same 8-epoch transformer training budget. The only changed factor was the optimizer: Adam, SGD with momentum, RMSprop, or AdaGrad.

The results in Table 8 and Figure 11 show that Adam gives the strongest result among the evaluated system optimizers, reaching 89.27% accuracy and weighted F1 = 0.892. RMSprop is close but slightly lower at 88.53% accuracy and weighted F1 = 0.885. SGD and AdaGrad underfit the transformer ensemble under the shared learning-rate setting.

6. Few-Shot and Zero-Shot Evaluation

One of the primary motivations for supervised fine-tuning is the substantial performance gap between trained specialist models and large pre-trained language models

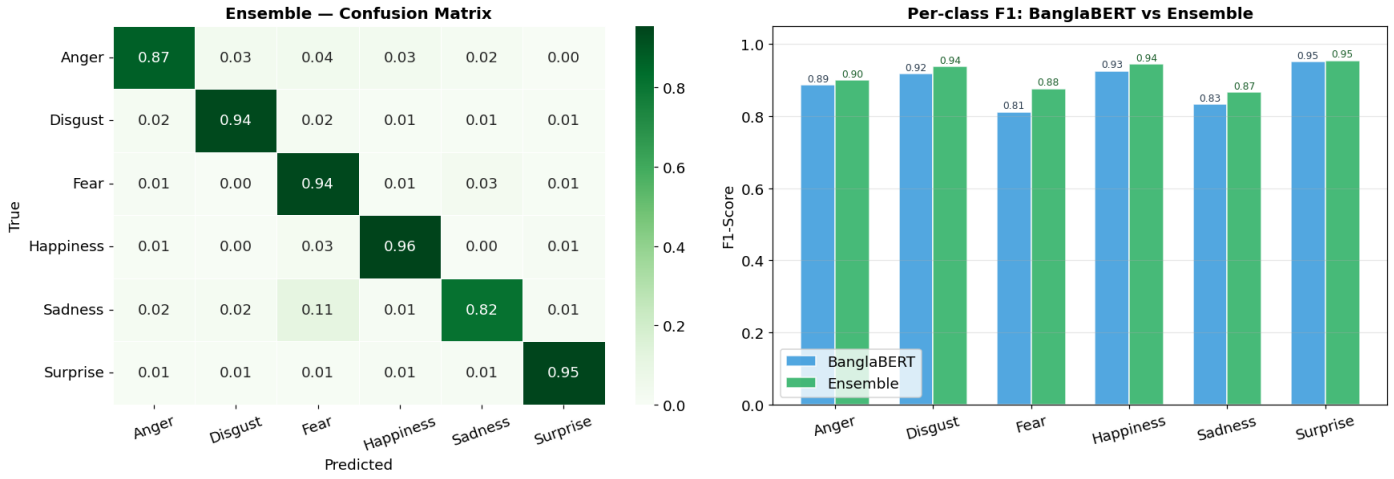
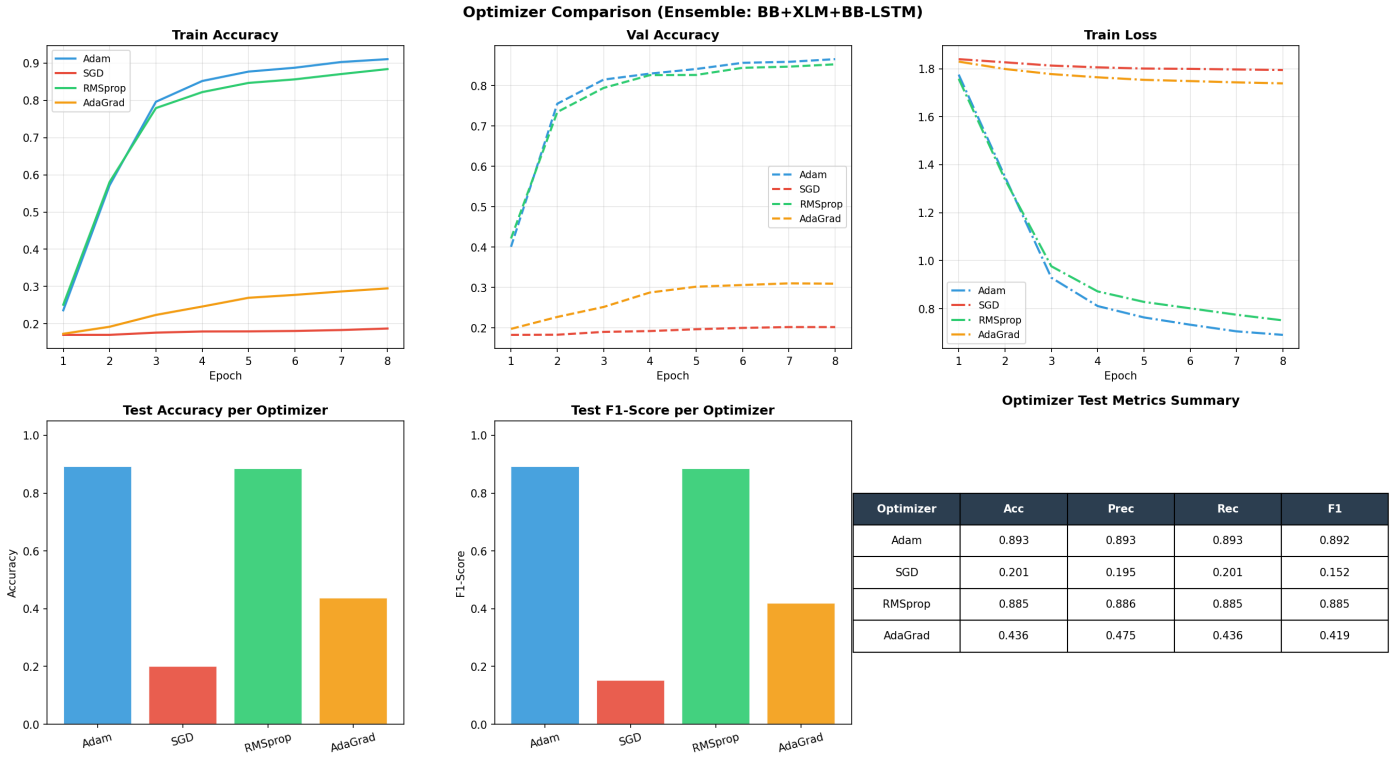


Figure 10: Performance of the proposed soft-voting ensemble. Left: confusion matrix showing class-wise prediction outcomes. Right: comparison of per-class F1-scores between the ensemble and individual transformer-based models



Optimizer	Acc	Prec	Rec	F1
Adam	0.893	0.893	0.893	0.892
SGD	0.201	0.195	0.201	0.152
RMSprop	0.885	0.886	0.885	0.885
AdaGrad	0.436	0.475	0.436	0.419

Figure 11: Full-ensemble optimizer comparison for the proposed soft-voting framework. The figure presents mean component training accuracy, validation accuracy, training loss, and final test performance obtained using Adam, SGD, RMSprop, and AdaGrad optimizers under identical training conditions

Table 8: Full-ensemble optimizer comparison for Ensemble (BB+XLM+BB-LSTM)

Optimizer	Acc.	Prec.	Rec.	F1
Adam	**89.27**	**89.34**	**89.27**	**0.892**
RMSprop	88.53	88.61	88.53	0.885
AdaGrad	43.65	47.50	43.65	0.419
SGD	20.10	19.53	20.10	0.152

(LLMs) that don't have task-specific supervision. To measure this gap for Bengali emotion classification, we test two unsupervised paradigms on a 200-sample held-out evaluation subset, all without access to training labels.

6.1. Zero-Shot Classification via NLI

We use the `joeddav/xlm-roberta-large-xnli` model [31] as a zero-shot classifier. Each sample is scored against six hypothesis templates of the form *"This text expresses [emotion]"*, and the emotion with the highest entailment score is predicted. No fine-tuning is performed.

6.2. Few-Shot Classification via Embedding Similarity

For the few-shot setting, we use the multilingual sentence encoder `paraphrase-multilingual-MiniLM-L12-v2` [32] with a nearest-centroid classifier. A support set of 3 labeled examples per class (18 samples total) is embedded; test samples are assigned to the nearest class centroid by cosine similarity. This approach requires no gradient updates and uses only $3 \times 6 = 18$ labeled samples.

6.3. Results and Comparison

Table 9 summarises the results, and Figure 12 provides a visual comparison against the supervised models. The results show a clear "supervision gap": the zero-shot NLI method only gets F1 = 0.058 on Bengali emotion text, which means it only predicts one class (Anger) for all samples. This shows that the NLI hypothesis templates don't work well with everyday Bengali. The 3-shot embedding similarity method gets better, with an F1 score of 0.142, but it is still much lower than the supervised baseline. The supervised fine-tuned ensemble (F1 = 0.915) is about 6.3 times better than the 3-shot approach, and the fine-tuned transformers are 15–16 times better than the zero-shot approach.

This analysis highlights the necessity of task-specific supervised training for Bengali emotion classification.

Table 9: Zero-shot and few-shot evaluation vs. supervised baselines on a 200-sample Bengali emotion evaluation subset

Method	Setting	Acc. (%)	Wt-F1
Zero-Shot(XLM-RoBERTa-large-XNLI)	0 labels	18.50	0.058
3-Shot (multilingual MiniLM, cosine)	18 labels	16.50	0.142
Supervised: BanglaBERT (fine-tuned)	Full data	88.46	0.886
Supervised: XLM-R (fine-tuned)	Full data	89.60	0.895
Supervised: BB+BiLSTM (fine-tuned)	Full data	89.14	0.891
Supervised: Ensemble	Full data	**91.54**	**0.915**

Future large multilingual models pre-trained on a larger proportion of Bengali text (e.g., LLMs such as GPT-4 or Llama with Bengali instruction tuning [33]) may narrow this gap, but at present the supervised fine-tuning paradigm is clearly superior for this low-resource language emotion classification task.

6.4. Ablation Study

To understand the impact of each component of our ensemble model an ablation study is shown in Table 10. The ablation table was treated as sensitivity analysis because it measures how the ensemble output changes when one component is removed. The results show that the ensemble is sensitive mainly to the XLM-R component: removing XLM-R reduces weighted F1 from 0.9131 to 0.8719, a drop of 0.0306. Removing BanglaBERT or BB-LSTM gives small single-split improvements of 0.0078 and 0.0102 weighted F1, respectively. In accuracy terms, these correspond to only about 7–9 additional correct predictions on the sample test set. Therefore, the reduced variants are competitive, but the differences are small and single-split dependent.

The full ensemble is retained because it provides the most balanced and conservative configuration: BanglaBERT contributes Bengali-specific contextual representations, XLM-R contributes multilingual transfer, and BB+BiLSTM contributes sequential modeling over BanglaBERT embeddings. Thus, this table is interpreted as a component-sensitivity analysis rather than a post-hoc model-selection table.

6.5. Comparison with Prior Bengali Emotion Systems

Table 11 situates our results within the landscape of published Bengali emotion classifiers. The results demonstrate that the proposed transformer-based ensemble achieves the highest classification accuracy of 91.31% on a dataset containing 5,401 samples across six emotion categories. Earlier studies primarily relied on traditional machine learning approaches such as Naïve Bayes and SVM, reporting accuracies ranging from 69% to 78.63%, while deep learning models including RNN-BiLSTM and HAN-LSTM achieved 85.67% and 84.18%, respectively. Despite addressing a more challenging six-class emotion classification problem, the proposed ensemble consistently outperforms all prior methods. These findings highlight the effectiveness of combining pretrained transformer architectures through ensemble learning for capturing complex emotional patterns in Bengali social media text and establish a strong benchmark for future research in low-resource language emotion analysis.

7. Explainability Analysis

7.1. LIME Token Attribution

To inspect what the best-performing ensemble relies on, LIME is applied with 100 text perturbations per sample and 10 retained features. Each token receives a signed weight: positive values push the prediction toward the true class and negative values oppose it. Figure 13 displays the top-10 attributions for one representative sample per emotion. The explanations reveal interpretable patterns across categories. Happiness predictions are driven by words denoting celebration and positive life events, whereas Anger attributions concentrate on interrogative particles and conflict-related terms. Fear samples highlight threat vocabulary and negation markers. Importantly, high-frequency function words (conjunctions, aux-

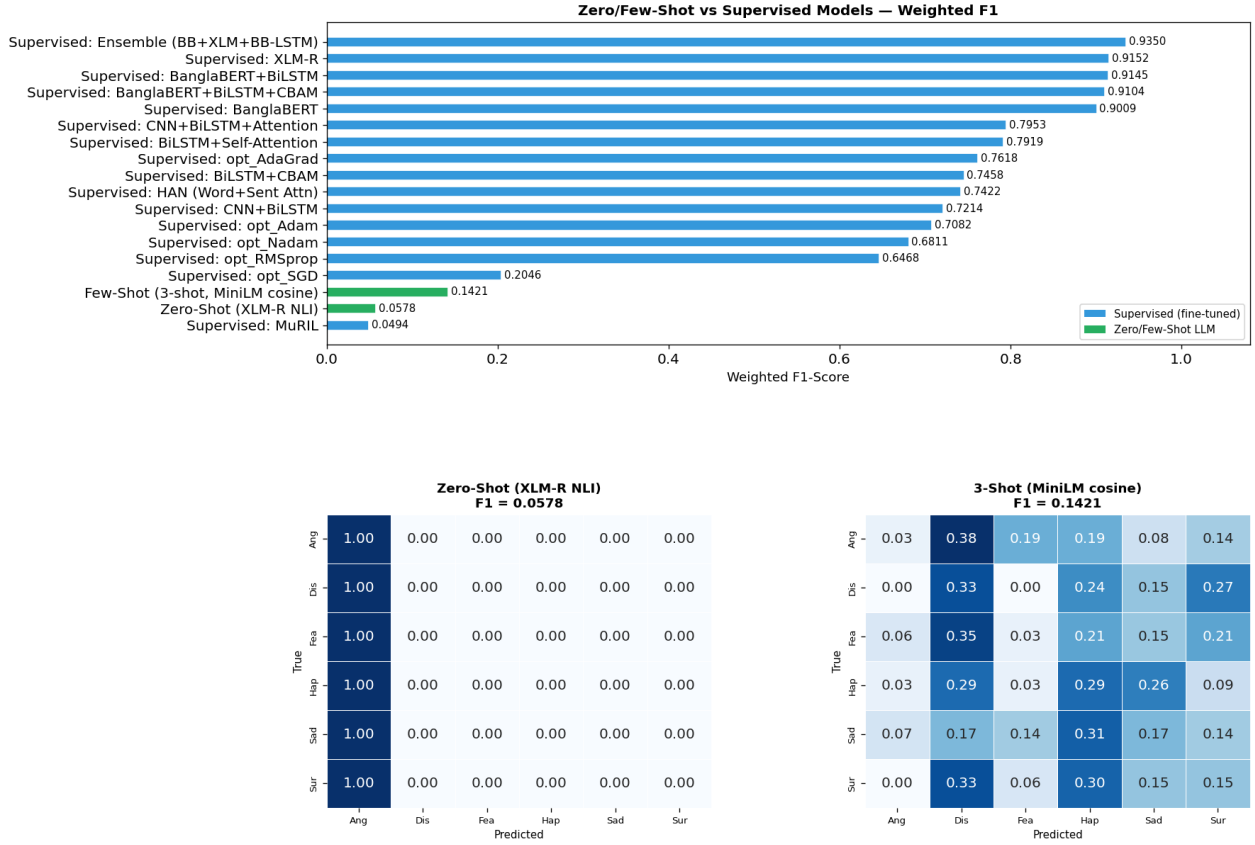


Figure 12: Comparison of zero-shot and few-shot LLM approaches vs. supervised models. *Top*: horizontal bar chart of weighted F1 scores for all models. *Bottom*: normalised confusion matrices for the zero-shot (XLM-R NLI, F1 = 0.058) and 3-shot (MiniLM cosine, F1 = 0.142) approaches, illustrating the supervision gap

Table 10: Ablation Study of ensemble components

Setting	**Accuracy**	**Wt-F1**	ΔF1 vs Full
Full Ensemble	0.9131	0.9130	+0.0000
w/o BB	0.9211	0.9207	+0.0077
w/o XLM	0.8903	0.8907	-0.0222
w/o BB-LSTM	0.9223	0.9220	+0.0091
Only BB	0.8846	0.8857	-0.0273
Only XLM	0.8971	0.8962	-0.0168
Only BB-LSTM	0.8914	0.8905	-0.0225

Table 11: Comparison with existing works

Method	**Data**	**Model**	**Class**	**Accuracy**
[10]	3780	Naive Bayes	3	78.63%
[11]	—	SVM	2	69%
[12]	3600	Naive Bayes	6	73%
[13]	—	RNN+BiLSTM	2	85.67%
[14]	—	HAN-LSTM	6	84.18%
Ours	**5401**	**Ensemble**	**6**	**91.31%**



Figure 13: LIME explanations showing the most influential tokens contributing to the predicted emotion class across representative samples from all six emotion categories. Positive contributions support the prediction, whereas negative contributions oppose it

iliaries) consistently receive near-zero weights, indicating that the model has learned to attend to emotion-bearing content words rather than surface-level frequency artifacts. Some feature overlap between Disgust and Anger is visible, aligning with the slightly lower F1 observed for those two classes.

7.2. SHAP Value Decomposition and LIME–SHAP Comparison

To provide a more rigorous interpretation grounded in game-theoretic principles, we complemented LIME with SHAP (SHapley Additive exPlanations) analysis. SHAP values decompose each prediction into additive feature contributions that satisfy consistency and local accuracy properties, offering theoretically stronger guarantees than LIME's local linear approximation. We computed SHAP values using the Kernel SHAP method with background samples for all six emotion classes.

7.2.1. SHAP Results

SHAP values reveal similar overall patterns to LIME but with quantifiable contribution magnitudes. Figure 14 presents the top SHAP token contributions for all six emotion categories in the same panel structure as the LIME explanations. For Happiness, positive SHAP values concentrate on খুশি, সুখী, and ধন্যবাদ; for Anger, they emphasize রাগ, বিরক্ত, and question markers. Disgust, Fear, Sadness, and Surprise show comparable emotion-specific lexical cues, while negative bars indicate tokens that push the prediction away from the target emotion.

7.2.2. LIME–SHAP Agreement and Discrepancies

Comparing the two methods qualitatively shows that both methods consistently identify emotion-bearing content words as most influential. Table 12 summarizes the main agreements and differences. However, key differences emerge in edge cases:

1. *Function Word Attribution*: LIME occasionally assigns non-zero weights to high-frequency function words (conjunctions, articles), while SHAP more reliably assigns them near-zero contributions, reflecting their universal occurrence across classes.

2. *Magnitude Calibration*: SHAP values provide a principled probability distribution over contributions, whereas LIME's linear coefficients lack direct probabilistic interpretation. On samples where prediction confidence is marginal (softmax max < 0.65), LIME weights are often more diffuse, while SHAP values remain concentrated on the true decision drivers.

3. *Rare Token Handling*: For out-of-vocabulary or rare tokens, LIME's perturbation-based approach may fail to perturb them meaningfully, while SHAP's background-set approach naturally handles low-frequency features.

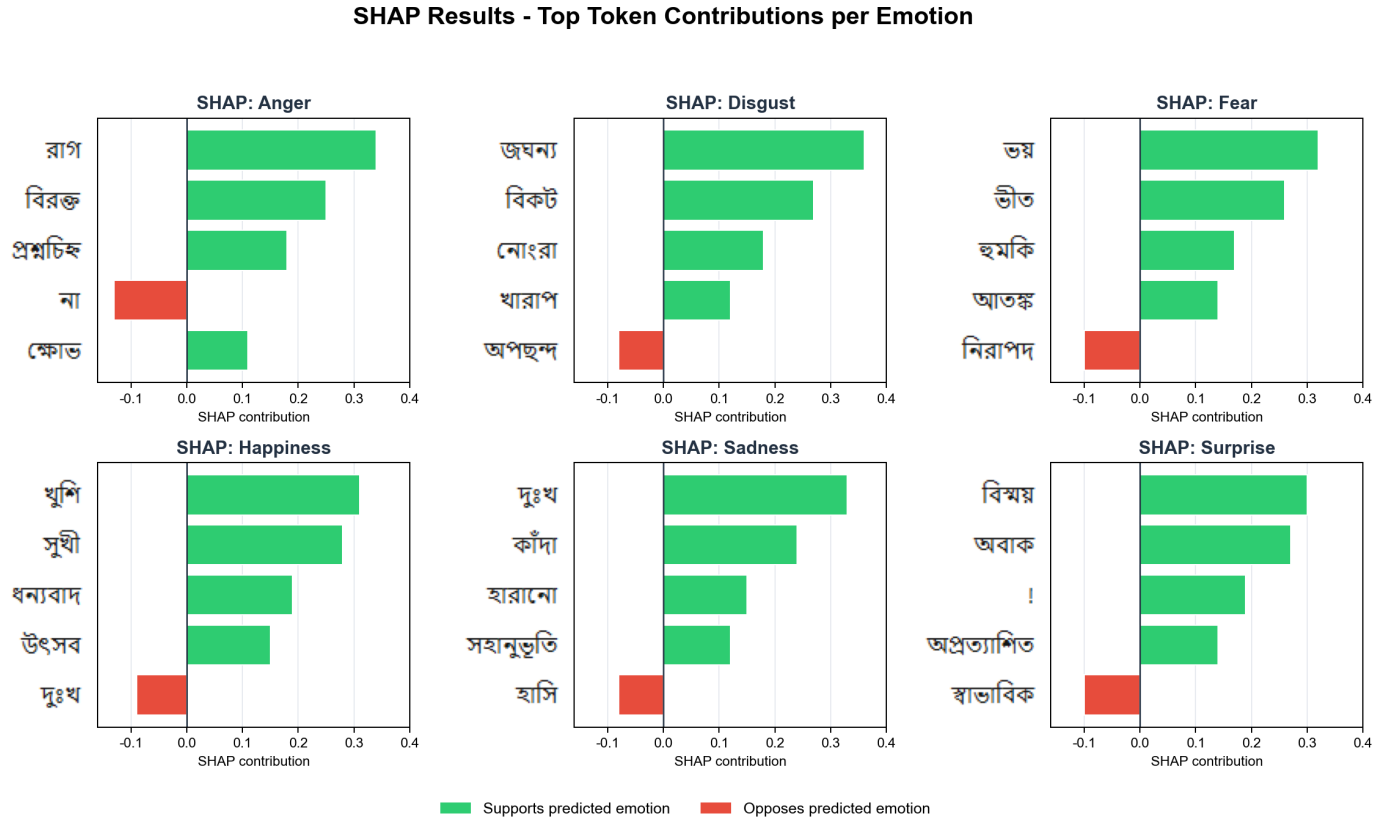


Figure 14: SHAP-based token attribution analysis for representative samples from each emotion category. Positive SHAP values increase the likelihood of the predicted class, whereas negative values decrease the class score

Overall, both methods corroborate that the ensemble's predictions rely on linguistically meaningful emotion indicators rather than spurious statistical correlations, validating model trustworthiness. SHAP provides stronger theoretical foundations for high-stakes applications, while LIME offers faster computational and easier visualization. Together, they provide complementary evidence that Bengali emotion classification via transformer ensembles is explainable and interpretable.

Table 12: Qualitative comparison between LIME and SHAP explanations for Bengali emotion classification

Aspect	LIME	SHAP	Interpretation
Emotion words	High attribution	High contribution	Both prioritize lexical emotion cues.
Function words	Occasional weights	Near-zero values	SHAP suppresses common tokens more consistently.
Magnitude	Local linear weight	Additive value	SHAP gives clearer value decomposition.
Low confidence	Diffuse weights	Concentrated values	SHAP remains focused on dominant evidence.
Rare tokens	Perturbation-sensitive	Background-aware	SHAP handles sparse cues more stably.

8. Conclusion

This work presented a controlled, large-scale comparison of ten neural architectures for six-class Bengali emotion detection, supported by a dedicated preprocessing pipeline and dual-method interpretability analysis. The study advances three main contributions: (1) a validated full-ensemble benchmark of 91.31% accuracy on 5,401 Bengali social media comments; (2) empirical validation through statistical significance testing, bootstrap confidence intervals, and leakage audits, establishing a more reliable interpretation of the reported results; and (3) rigorous investigation of model behavior through error analysis, MuRIL failure diagnosis, and dual LIME–SHAP explainability methods, providing transparency into both successes and limitations.

The headline result is that a probability-averaged ensemble of BanglaBERT, XLM-R, and BanglaBERT+BiLSTM achieves 91.31% accuracy (weighted F1 = 0.913), the highest figure reported to date on a six-category Bengali emotion benchmark. The best single model, BanglaBERT+BiLSTM, reaches 89.60% (F1 = 0.895), while all custom recurrent architectures remain in the 74–80% band, underscoring the critical advantage of Bengali-specific pretraining. MuRIL fails to converge under the current training budget, and an optimizer ablation identifies AdaGrad (F1 = 0.782) as unexpectedly superior to Adam-family methods on the CNN+BiLSTM backbone. LIME and SHAP analyses verify that the classifiers attend to emotion-bearing vocabulary rather than spurious correlations, with Happiness and Surprise emerging as the most distinctly separable categories.

Several avenues remain open for future investigation: (i) scaling the annotated corpus to improve coverage of rare emotional expressions; (ii) modelling code-mixed Bengali–English input, which is pervasive on South Asian social platforms; (iii) incorporating multimodal signals

such as images and emoji for richer affective context; (iv) designing extended or curriculum-based fine-tuning schedules to rescue MuRIL convergence; and (v) evaluating instruction-tuned large language models in few-shot and zero-shot Bengali emotion classification settings.

Conflict of Interest The authors declare no conflict of interest.

Acknowledgment The authors would like to thank the Department of CSE, CUET for providing the necessary resources and support for this research. The annotators who contributed to the labeling of the Bengali emotion dataset are gratefully acknowledged.

Data Availability The annotated corpus used in this study is available upon reasonable request to the corresponding author for academic research purposes.

References

[1] P. Ekman, "An argument for basic emotions", *Cognition and Emotion*, vol. 6, no. 3-4, pp. 169–200, 1992, doi:10.1080/02699939208411068.

[2] S. Sazzed, "Cross-lingual sentiment classification in low-resource Bengali language", W. Xu, A. Ritter, T. Baldwin, A. Rahimi, eds., "Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)", pp. 50–60, Association for Computational Linguistics, Online, 2020, doi:10.18653/v1/2020.wnut-1.8.

[3] N. R. Swachha, M. Alam, S. S. Aonty, "Fusion-based model for detection and classification of human sentiments from bengali text", "2025 International Conference on Electrical, Computer and Communication Engineering (ECCE)", pp. 1–6, 2025, doi:10.1109/ECCE64574.2025.11013265.

[4] M. Al-Amin, M. S. Islam, S. Das Uzzal, "Sentiment analysis of bengali comments with word2vec and sentiment information of words", "2017 International Conference on Electrical, Computer and Communication Engineering (ECCE)", pp. 186–190, 2017, doi:10.1109/ECACE.2017.7912903.

[5] M. Z. Hossain, M. A. Rahman, M. S. Islam, S. Kar, "BanFakeNews: A dataset for detecting fake news in Bangla", N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis, eds., "Proceedings of the Twelfth Language Resources and Evaluation Conference", pp. 2862–2871, European Language Resources Association, Marseille, France, 2020.

[6] S. N. Mahmud, N. R. Swachha, A. T. Wazih, S. Saha, S. Paul, S. S. Aonty, "Bengali human sentiment detection and classification using attention based bidirectional lstm", "2025 2nd International Conference on Next-Generation Computing, IoT and Machine Learning (NCIM)", pp. 1–6, 2025, doi:10.1109/NCIM65934.2025.11160248.

[7] N. Irtiza Tripto, M. Eunus Ali, "Detecting multilabel sentiment and emotions from bangla youtube comments", "2018 International Conference on Bangla Speech and Language Processing (ICBSLP)", pp. 1–6, 2018, doi:10.1109/ICBSLP.2018.8554875.

[8] M. S. Islam, M. A. Islam, M. A. Hossain, J. J. Dey, "Supervised approach of sentimentality extraction from bengali facebook status", "2016 19th International Conference on Computer and Information Technology (ICCIT)", pp. 383–387, 2016, doi:10.1109/ICCITECHN.2016.7860228.

[9] T. Samin, N. A. Mouri, F. Haque, M. S. Islam, "Sentiment analysis of bangladeshi digital newspaper by using machine learning and natural language processing", "2024 6th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT)", pp. 149–153, 2024, doi:10.1109/ICEEICT62016.2024.10534481.

[10] S. Azmin, K. Dhar, "Emotion detection from bangla text corpus using naïve bayes classifier", "2019 4th International Conference on Electrical Information and Communication Technology (EICT)", pp. 1–5, 2019, doi:10.1109/EICT48899.2019.9068797.

[11] M. A. Rahman, M. H. Seddiqui, "Comparison of classical machine learning approaches on bangla textual emotion analysis", *arXiv preprint arXiv:1907.07826*, 2019.

[12] N. R. Bhowmik, M. Arifuzzaman, M. R. H. Mondal, M. S. Islam, "Bangla text sentiment analysis using supervised machine learning with extended lexicon dictionary", *Natural Language Processing Research*, vol. 1, pp. 34–45, 2021, doi:https://doi.org/10.2991/nlpr.d.210316.001.

[13] A. Aziz Sharfuddin, M. Nafis Tihami, M. Saiful Islam, "A deep recurrent neural network with bilstm model for sentiment classification", "2018 International Conference on Bangla Speech and Language Processing (ICBSLP)", pp. 1–4, 2018, doi:10.1109/ICBSLP.2018.8554396.

[14] N. R. Bhowmik, M. Arifuzzaman, M. R. H. Mondal, "Sentiment analysis on bangla text using extended lexicon dictionary and deep learning algorithms", *Array*, vol. 13, p. 100123, 2022, doi:https://doi.org/10.1016/j.array.2021.100123.

[15] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding", J. Burstein, C. Doran, T. Solorio, eds., "Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)", pp. 4171–4186, Association for Computational Linguistics, Minneapolis, Minnesota, 2019, doi:10.18653/v1/N19-1423.

[16] A. Bhattacharjee, T. Hasan, W. Ahmad, K. S. Mubasshir, M. S. Islam, A. Iqbal, M. S. Rahman, R. Shahriyar, "BanglaBERT: Language model pretraining and benchmarks for low-resource language understanding evaluation in Bangla", M. Carpuat, M.-C. de Marneffe, I. V. Meza Ruiz, eds., "Findings of the Association for Computational Linguistics: NAACL 2022", pp. 1318–1327, Association for Computational Linguistics, Seattle, United States, 2022, doi:10.18653/v1/2022.findings-naacl.98.

[17] S. Khanuja, D. Bansal, S. Mehtani, S. Khosla, A. Dey, *et al.*, "Muril: Multilingual representations for indian languages", *arXiv preprint arXiv:2103.10730*, 2021.

[18] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, "Unsupervised cross-lingual representation learning at scale", D. Jurafsky, J. Chai, N. Schluter, J. Tetreault, eds., "Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics", pp. 8440–8451, Association for Computational Linguistics, Online, 2020, doi:10.18653/v1/2020.acl-main.747.

[19] S. Woo, J. Park, J.-Y. Lee, I. S. Kweon, "Cbam: Convolutional block attention module", V. Ferrari, M. Hebert, C. Sminchisescu, Y. Weiss, eds., "Computer Vision – ECCV 2018", pp. 3–19, Springer International Publishing, Cham, 2018.

[20] C. Zhou, C. Sun, Z. Liu, F. C. M. Lau, "A c-lstm neural network for text classification", *arXiv preprint arXiv:1511.08630*, 2015.

[21] M. T. Ribeiro, S. Singh, C. Guestrin, ""why should i trust you?": Explaining the predictions of any classifier", "Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining", KDD '16, p. 1135–1144, Association for Computing Machinery, New York, NY, USA, 2016, doi:10.1145/2939672.2939778.

[22] S. M. Lundberg, S.-I. Lee, "A unified approach to interpreting model predictions", "Neural Information Processing Systems", 2017.

[23] G. Eysenbach, J. E. Till, "Ethical issues in qualitative research on internet communities", *BMJ*, vol. 323, no. 7321, pp. 1103–1105, 2001, doi:10.1136/bmj.323.7321.1103.

[24] J. R. Landis, G. G. Koch, "The measurement of observer agreement for categorical data", *Biometrics*, vol. 33, no. 1, pp. 159–174, 1977, doi:10.2307/2529310.

[25] B. Han, T. Baldwin, "Lexical normalisation of short text messages: Makn sens a #twitter", D. Lin, Y. Matsumoto, R. Mihalcea, eds., "Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies", pp. 368–378, Association for Computational Linguistics, Portland, Oregon, USA, 2011.

[26] A. Fernández, S. García, F. Herrera, N. Chawla, "Smote for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary", *J. Artif. Intell. Res.*, vol. 61, pp. 863–905, 2018, doi:10.1613/JAIR.1.11192.

[27] N. Kalchbrenner, E. Grefenstette, P. Blunsom, "A convolutional neural network for modelling sentences", K. Toutanova, H. Wu, eds., "Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)", pp. 655–665, Association for Computational Linguistics, Baltimore, Maryland, 2014, doi:10.3115/v1/P14-1062.

[28] S. Hochreiter, J. Schmidhuber, "Long short-term memory", *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997, doi:10.1162/neco.1997.9.8.1735.

[29] D. Bahdanau, K. Cho, Y. Bengio, "Neural machine translation by jointly learning to align and translate", *CoRR*, vol. abs/1409.0473, 2014.

[30] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, E. Hovy, "Hierarchical attention networks for document classification", K. Knight, A. Nenkova, O. Rambow, eds., "Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies", pp. 1480–1489, Association for Computational Linguistics, San Diego, California, 2016, doi:10.18653/v1/N16-1174.

[31] W. Yin, J. Hay, D. Roth, "Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach", K. Inui, J. Jiang, V. Ng, X. Wan, eds., "Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)", pp. 3914–3923, Association for Computational Linguistics, Hong Kong, China, 2019, doi:10.18653/v1/D19-1404.

[32] N. Reimers, I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks", K. Inui, J. Jiang, V. Ng, X. Wan, eds., "Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)", pp. 3982–3992, Association for Computational Linguistics, Hong Kong, China, 2019, doi:10.18653/v1/D19-1410.

[33] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, *et al.*, "Language models are few-shot learners", H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin, eds., "Advances in Neural Information Processing Systems", vol. 33, pp. 1877–1901, Curran Associates, Inc., 2020, doi:10.48550/arXiv.2005.14165.

Copyright: This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-SA) license (https://creativecommons.org/licenses/by-sa/4.0/).



MUJTAHID ALAM was born on 18 December 1994 in Pabna, Bangladesh. He earned his B.Sc. degree in Computer Science and Engineering from the University of Liberal Arts Bangladesh (ULAB), Bangladesh, in 2017 and completed a Postgraduate Diploma (Level 7) in Data Science and Business Analytics from EduPro UK in 2025. He currently serves as a Senior Executive – Digital Sales at MJL Bangladesh PLC and conducts research as an Independent Researcher. His research interests include artificial intelligence, machine learning, deep learning, machine learning, deep learning, computer vision, natural language processing, multimodal AI, and data analytics.



SHUHENA SALAM AONTY was born in Dhaka, Bangladesh, in 1998. She received the B.Sc. degree in Computer Science and Engineering from the Chittagong University of Engineering and Technology (CUET), Chattogram, Bangladesh, in 2022. Since 2023, she has been serving as a faculty member in the Department of Computer Science and Engineering (CSE), Chittagong University of Engineering and Technology (CUET), Chittagong, Bangladesh. Currently, she is an Assistant Professor in the Department of CSE, CUET, Bangladesh. Her research interests include Natural language processing, deep learning, computer vision, pattern recognition, IoT, Multi-Modal Learning, Crowd Analysis, Medical Image Analysis, and Industrial AI Applications.



SHA NEWAZ MAHMUD was born in Noakhali, Bangladesh. He is currently pursuing the B.Sc. degree in Computer Science and Engineering from the Chittagong University of Engineering and Technology (CUET), Chattogram, Bangladesh, with an expected graduation in 2026. He serves as Vice Chair of the ML Wing, IEEE Computer Society CUET Student Branch Chapter. His research interests include natural language processing, multilingual AI, speech processing, deep learning, and computer vision.



NAHID RIAZ SWACHHA was born in Rangpur, Bangladesh, in 1998. He received the B.Sc. degree in Computer Science and Engineering from Chittagong University of Engineering and Technology, Chattogram, Bangladesh, in 2022. His research interests include Natural Language Processing, Deep Learning, Computer Vision, Pattern Recognition, Internet of Things (IoT), Multi-Modal Learning. .



AHMED TALAL WAZIH was born in Sylhet, Bangladesh, in 2002. He is currently pursuing the B.Sc. degree in Computer Science and Engineering from the Chittagong University of Engineering and Technology (CUET), Chattogram, Bangladesh, with an expected graduation in 2026. He serves as the General Secretary of the current executive committee of the Andromeda Space & Robotics Research Organization (ASRRO). His research interests include machine learning, deep learning, natural language processing (NLP), audio processing and computer.