# Zero- and Few-Shot Prompting with LLMs: A Comparative Study with Fine-tuned Models for Bangla Sentiment Analysis

**Md. Arid Hasan**[1], **Shudipta Das**[2], **Afiyat Anjum**[2], **Firoj Alam**[3],
**Anika Anjum**[2], **Avijit Sarker**[2], **Sheak Rashed Haider Noori**[2]
[1]SE+AI Research Lab, University of New Brunswick, Fredericton, Canada
[2]Daffodil International University, Dhaka, Bangladesh
[3]Qatar Computing Research Institute, Doha, Qatar
arid.hasan@unb.ca, fialam@hbku.edu.qa

## Abstract

The rapid expansion of the digital world has propelled sentiment analysis into a critical tool across diverse sectors such as marketing, politics, customer service, and healthcare. While there have been significant advancements in sentiment analysis for widely spoken languages, low-resource languages, such as Bangla, remain largely under-researched due to resource constraints. Furthermore, the recent unprecedented performance of Large Language Models (LLMs) in various applications highlights the need to evaluate them in the context of low-resource languages. In this study, we present a sizeable manually annotated dataset encompassing 33,606 Bangla news tweets and Facebook comments. We also investigate zero- and few-shot in-context learning with several language models, including Flan-T5, GPT-4, and Bloomz, offering a comparative analysis against fine-tuned models. Our findings suggest that monolingual transformer-based models consistently outperform other models, even in zero and few-shot scenarios. To foster continued exploration, we intend to make this dataset and our research tools publicly available to the broader research community.

**Keywords:** Sentiment, LLMs, Zero-shot, Few-shot, Bangla NLP, Low-resource language

## 1. Introduction

Sentiment analysis is an influential sub-area of NLP that deals with sentiment, emotions, affect and stylistic analysis in language. There has been significant research effort for sentiment analysis due to its need in various fields, such as business, finance, politics, education, and services (Cui et al., 2023). The analysis typically has been done on different types of content – domains (news, blog posts, customer reviews, social media posts), modalities (textual and multimodal) (Hussein, 2018; Dashtipour et al., 2016). The surge in user-generated content on social media platforms has become a significant phenomenon, as individuals increasingly voice their opinions on a wide array of topics through comments and tweets. As a result, these platforms have garnered considerable research attention as valuable sources of data for sentiment analysis (Yue et al., 2019). Leveraging such data resources (Dashtipour et al., 2016), substantial progress has been achieved for the sentiment analysis in English. The advancements range from quantifying sentiment polarity to tackling more complex challenges like identifying aspects (Chen et al., 2022), multimodal sentiment detection (Liang et al., 2022), explainability (Cambria et al., 2022), and multilingual sentiment analysis (Barbieri et al., 2022; Galeshchuk et al., 2019).

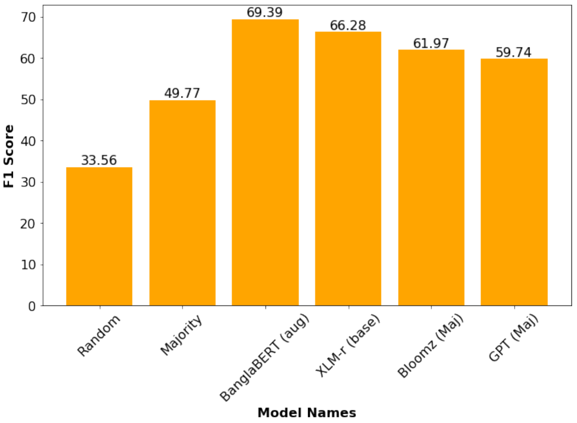

Figure 1: Performance comparisons with baselines (random and majority), fine-tuned models and LLMs (GPT and Bloomz).

There has been a growing research interest over time in sentiment analysis for low-resource languages (Batanović et al., 2016; Nabil et al., 2015; Muhammad et al., 2023). Similar to other low-resource languages, the research for the sentiment analysis for Bangla has been limited (Islam et al., 2021, 2023). A study conducted by Alam et al. (2021a) emphasized the primary challenges associated with Bangla sentiment analysis, specifically issues of duplicate instances in the data, inadequate reporting of annotation agreement, and generalization. These challenges were also highlighted in (Islam et al., 2021), further emphasizing the need to address them for effective sentiment analysis in Bangla. To further facilitate sentiment

*LREC-COLING 2024*, pages 17808–17818
20-25 May, 2024. © 2024 ELRA Language Resource Association: CC BY-NC 4.0

analysis research in Bangla, we have created a multi-platform sentiment analysis dataset in this study. The dataset has undergone multiple rounds of pre-processing and validation to ensure its suitability for both sentiment analysis tasks and qualitative investigations.

We provide a comparative analysis using various pre-trained language models, including zero and few-shot settings with different fine-tuned models and LLMs as presented in Figure 1. The analysis definitively demonstrates that LLMs surpass both random and majority baselines in performance, yet they fall short when compared to fine-tuned models. More details are discussed in the *Results* section.

Our contributions can be summarized as follows:

- Development of one of the largest manually annotated datasets for sentiment analysis.
- Investigation into zero-shot and few-shot learning using various LLMs. We are the first to provide such a comprehensive evaluation for Bangla sentiment analysis.
- Comparative analysis of performance differences between in-context learning and fine-tuned models.
- Investigation of how different prompting variations affect performance in in-context learning.

Based on our extensive experiments our findings are summarized below:

- Fine-tuned models yield better results compared to both zero- and few-shot in-context learning setups.
- Fine-tuned models using monolingual text (BanglaBERT) demonstrate superior performance.
- There is little to no performance difference between zero- and few-shot learning with the GPT-4 model.
- For the majority of zero- and few-shot experiments, BLOOMZ yielded better performance than GPT-4.
- While BLOOMZ failed to predict the neutral class, GPT-4 struggled with positive class prediction.

The remainder of the paper is structured as follows: Section *Related Work* provides an overview of relevant literature. The *Dataset* section provides the details of the dataset used, along with an analysis of its contents. In Section *Methodology*, we discuss the models and experiments. The *Results and Discussion* section presents and discusses our findings. Lastly, Section *Conclusion* provides concluding remarks.

## 2. Related Work

In the realm of sentiment classification for Bangla, the current state-of-the-art research focuses on two key aspects: resource development and tackling model development challenges. Earlier work in this area has encompassed rule-based methodologies as well as classical machine learning approaches and recently the use of pre-trained models has received a wider attention.

### 2.1. Datasets

Over time there have been several resources developed including manual and semi-supervised labeling approaches (Chowdhury and Chowdhury, 2014; Alam et al., 2021a; Islam et al., 2021, 2023; Kabir et al., 2023). Chowdhury and Chowdhury (2014) developed a dataset using semi-supervised approaches and trained models SVM and Maximum Entropy. The study of Kabir et al. (2023) proposed an annotated sentiment corpus comprising 158,065 reviews collected from online bookstores. The annotations were primarily based on the rating of the reviews, with the majority (89.6%) being in the positive class. The study also evaluated classical and BERT-based models for training and performance assessment. The skewness of this dataset makes it particularly challenging. SentiGold (Islam et al., 2023)[1] is a well-balanced sentiment dataset containing 70K entries from 30 different domains. It was collected from various sources, including YouTube, Facebook, newspapers, blogs, etc., and labeled into five classes. The reported inter-annotator agreement is 0.88.

Rahman and Kumar Dey (2018) labeled 5,700 instances as positive, negative, or neutral for two aspect-based sentiment analysis (ABSA) tasks, specifically extracting aspect categories and polarity. The authors curated two new datasets from the cricket and restaurant domains. Islam et al. (2021) developed the SentNoB dataset, comprising 15,000 manually annotated comments collected from the comments section of Bangla news articles and videos across 13 diverse domains. The experimental findings using this dataset indicate that lexical feature combinations outperform neural models.

### 2.2. Models

Various classical algorithms have been employed in different studies for sentiment classification in Bangla. These include Bernoulli Naive Bayes (BNB), Decision Tree, Support Vector Machine (SVM), Maximum Entropy (ME), and Multinomial Naive Bayes (MNB) (Rahman and Hossen, 2019;

[1] Note that this dataset is not publicly available.

Banik and Rahman, 2018; Chowdhury et al., 2019). Islam et al. (2016) developed a sentiment classification system for textual movie reviews in Bangla. The authors utilized two machine learning algorithms, Naive Bayes (NB) and SVM, and provided comparative results. Additionally, Islam et al. (2016) employed NB with rules for sentiment detection in Bengali Facebook statuses.

Deep learning algorithms have been extensively explored in the context of Bangla sentiment analysis (Hassan et al., 2016; Aziz Sharfuddin et al., 2018; Tripto and Ali, 2018; Ashik et al., 2019; Karim et al., 2020; Sazzed, 2021; Sharmin and Chakma, 2021). In the study conducted by Tripto and Ali (2018), the authors utilized Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNNs) with an embedding layer to identify both sentiment and emotion in YouTube comments. Ashik et al. (2019) conducted a comparative analysis of classical algorithms, such as Support Vector Machines (SVM), alongside deep learning algorithms, including LSTM and CNN, for sentiment classification of Bangla news comments. Karim et al. (2020) integrated word embeddings into a Multichannel Convolutional-LSTM (MConv-LSTM) network, enabling the prediction of various types of hate speech, document classification, and sentiment analysis in the Bangla language. Another aspect explored in sentiment analysis is the utilization of LSTM models due to the prevalence of romanized Bangla texts in social media. Hassan et al. (2016); Aziz Sharfuddin et al. (2018) employed LSTM models to design and evaluate their sentiment analysis models, taking into account the unique characteristics of romanized Bangla texts.

In the study conducted by Hasan et al. (2020), a comprehensive comparison was performed on various annotated sentiment datasets consisting of Bangla content from social media sources. The research investigated the effectiveness of both classical algorithms, such as SVM, and deep learning algorithms, including CNN and transformer models. Notably, the deep learning algorithm XLMRoBERTa exhibited superior performance with an accuracy of 0.671, surpassing the classical algorithm SVM, which achieved an accuracy of 0.581.

In a review article by Alam et al. (2021a), the authors investigated nine NLP tasks, including sentiment analysis. They reported that transformer-based models, particularly XLM-RoBERTa-large, are more suitable for Bangla text categorization problems than other machine learning techniques such as LSTM, BERT, and CNN.

**Our Study:** We developed the largest MUltiplatform BAngla SEntiment (MUBASE) social media dataset, consisting of Facebook posts and tweets. Following the recommendations outlined in Alam et al. (2021a), we ensured that the dataset is clean, free of duplicates, and possesses high annotation quality with an annotation agreement score of $\kappa$=0.84. We have made the dataset publicly available[2] to the community. We conducted experiments that go beyond traditional approaches and smaller transformer-based models. Specifically, we investigated the effectiveness of advanced models such as Flan-T5, GPT-4, and BLOOMZ in both zero- and few-shot settings.

# 3. Dataset

## 3.1. Data Collection

We collected tweets and comments from both Facebook posts and Twitter. To collect tweets, we focused on user accounts associated with the following news media sources: BBC Bangla, Prothom Alo, and BD24Live. For the comments from the Facebook posts, we selected public pages belonging to several news media outlets. Our selection of news media was based on the availability of a substantial number of comments. In total, we collected approximately 35,000 posts/comments associated with various Bangla news portals. Then we removed all the posts, which contains only emojis and URLs as well as duplicate data and filtered tweets while collecting through API. We also removed all the Banglish (Bangla text written in English alphabets) comments from our initial dataset. These filtering and duplicate-removal steps resulted in $33,606$ entries. In the rest of the paper, we will use the term post to refer to posts and comments.

Table 1 presents the distribution of the number of posts and comments associated with each social media source. Our preliminary study reveals that Twitter users post both positive and negative sentiments, while showing fewer neutral expressions. On the other hand, Facebook users post more negative sentiments. Overall, the distribution of posts with negative sentiment is higher in the dataset. We further analyzed the distribution of sentences by the number of words associated with each class label, as shown in Figure 2. We created different ranges of sentence length buckets in order to understand and define the sequence length while training the transformer based models. It appears that more than 80% of the posts lie within twenty words, which is expected with social media posts, as observed in previous studies (Alam et al., 2021b).

## 3.2. Annotation

To perform the annotation, we developed an annotation guideline based on previous studies (Mukta

[2] https://github.com/AridHasan/MUBASE

| Class | Facebook | Twitter | Total |
|---|---|---|---|
| Positive | 2,245 | 8,315 | 10,560 |
| Neutral | 4,866 | 1,331 | 6,197 |
| Negative | 9,078 | 7,771 | 16,849 |
| **Total** | 16,189 | 17,417 | 33,606 |

Table 1: Class label distribution across different sources of the dataset.

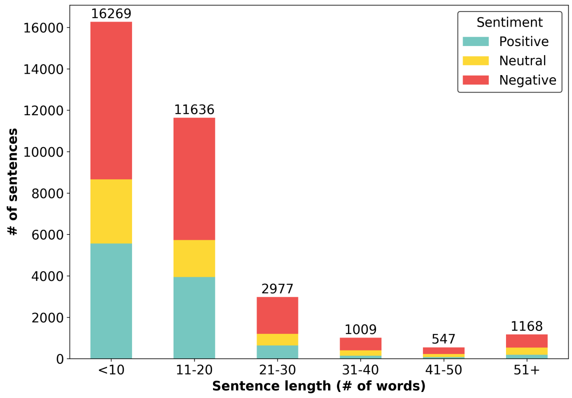

Figure 2: The distribution sentence length (number of words) associated with each sentiment label.

et al., 2021). For the Bangla sentiment polarity annotation, Mukta et al. (2021) proposed a classification with five labels: strongly negative, weakly negative, neutral, strongly positive, and weakly positive. However, due to the difficulty in distinguishing between strong and weak labels, we opted for a simplified approach with three labels: negative, neutral, and positive.

Each post was independently annotated by three annotators, all of whom are native speakers of Bangla. The annotators consisted of both male and female undergraduate students studying computer science. The final label for each post was determined based on the majority agreement among the annotators. However, in cases where there was disagreement among the annotators, a consensus meeting was organized to resolve any discrepancies and reach a final decision. Note that annotators are also the authors of the paper, hence, there has not been any payment for the annotation.

**Inter-Annotation Agreement** The quality of the annotations was assessed by calculating the inter-annotator agreement. As mentioned previously, three annotators independently annotated each post, adhering to the provided annotation instructions. We calculated the Fleiss Kappa ($\kappa$) score and obtained a value of $0.84$, indicating a perfect agreement among the annotators.[3]

| Class | Train | Dev | Test | Total |
|---|---|---|---|---|
| Positive | 7,342 | 1,126 | 2,092 | 10,560 |
| Neutral | 4,319 | 601 | 1,277 | 6,197 |
| Negative | 11,811 | 1,700 | 3,338 | 16,849 |
| **Total** | 23,472 | 3,427 | 6,707 | 33,606 |

Table 2: Class label distribution of the dataset.

### 3.3. Data Split

For our experiments, we divided the dataset into training, development, and test sets, comprising 70%, 10%, and 20% of the data, respectively. To ensure a balanced class label distribution across the sets, we employed stratified sampling (Sechidis et al., 2011). The distribution of the data split is provided in Table 2. The distribution in the table indicates a skew towards negative instances, followed by positive and neutral instances, suggesting that any analysis or model training based on this dataset may need to consider this imbalance.

## 4. Methodology

### 4.1. Data Pre-processing

The content shared on social media is mostly noisy and includes emoticons, usernames, hashtags, URLs, invisible characters, and symbols. To clean the data, we removed the noisy portion (emoticons, usernames, hashtags, URLs, invisible characters, etc.) of the data. Then we applied tokenization and removed the stopwords from the data. Identifying usernames in Facebook posts is more challenging than in tweets. While tweets precede usernames with an '@' symbol, Facebook posts have no such distinguishing pattern. To address this, we removed English text from Facebook posts since most usernames are in English. However, for usernames in Bangla text, removal was challenging due to the absence of a consistent pattern or a comprehensive Bangla name dictionary.

### 4.2. Evaluation Measures

For the performance measure for all different experimental settings, we compute accuracy, and weighted precision, recall and F$_1$ score. We choose to use the weighted version of the metric as it takes into account class imbalance.

### 4.3. Training and Evaluation Setup

For all experiments, except for LLMs (as detailed below), we trained the models using the training set, fine-tuned the parameters with the development set, and assessed the model's performance on the

[3] Note that values of Kappa of and 0.81–1.0 correspond perfect agreement (Landis and Koch, 1977).

test set. For the LLMs, we accessed them through APIs.

## 4.4. Models

We conducted our experiments using classical models as well as both small and large language models. It is worth noting that we follow the definitions of 'small' and 'large' models discussed in (Zhao et al., 2023). The term 'LLMs' refers to models encompassing tens or hundreds of billions of parameters.

### 4.4.1. Baseline

As baselines, we used both a majority (i.e., the class with the highest frequency) and a random approach. These methods have been widely used as baseline techniques in numerous studies, for example, (Rosenthal et al., 2017).

### 4.4.2. Classical Models

While classical models such as SVM (Platt, 1998) and Random Forest (Breiman, 2001) have been widely used in prior studies and remain in use in many low-resource production settings, we also wanted to assess their performance. To prepare the data for these models, we transformed the text into a tf-idf representation. During our experiments with SVM and RF, we used standard parameter settings: *1)* used n-gram (1 to 5) and transformed them into TF-IDF, *2)* for SVM we used the value of C =1, *3)* and for the Random Forest we used number of trees as 200.

### 4.4.3. Small Language Models (SLMs)

Large-scale pre-trained transformer models (PLMs) have achieved state-of-the-art performance across numerous NLP tasks. In our study, we fine-tuned several of these models. These included the monolingual transformer model BanglaBERT (Bhattacharjee et al., 2022) and multilingual transformers such as multilingual BERT (mBERT) (Devlin et al., 2019), XLM-RoBERTa (XLM-r) (Conneau et al., 2020), BLOOMZ (560m and 1.7B parameters models) (Muennighoff et al., 2023). We used the Transformer toolkit (Wolf et al., 2020) for the experiment. Following the guidelines outlined in (Devlin et al., 2019), we fine-tuned each model using the default settings over three epochs. Due to instability, we performed ten reruns for each experiment using different random seeds, and we picked the model that performed best on the development set. We provided the details of the parameters settings in Appendix A.

### 4.4.4. GPT Embedding

For many downstream NLP tasks, embedding extracted from pre-trained models followed by finetuning a feed-forward network provided reasonable results and also a reasonable setup for a low-resource production setting. Hence, we wanted to see the performance of this setting. We first extract the embeddings using OpenAI's text-embedding-ada-002 model for each data split. We then finetune a feed-forward network on the embeddings extracted from the training set to train our model. Our feed-forward model utilizes the Rectified Linear Unit (ReLU) activation function. We have set the learning rate to 0.001 and the hidden layer size to 500. We validate our model using the validation set and finally, we evaluate the model using the test set.

### 4.4.5. Large Language Models (LLMs)

For the LLMs, we investigate their performance with in-context zero- and few-shot learning settings without any specific training. It involves prompting and post-processing of output to extract the expected content. Therefore, for each task, we experimented with a number of prompts, guided by the same instruction and format as recommended in the OpenAI Chat playground, and PromptSource (Bach et al., 2022). We used the following models: Flan-T5 (large and XL) (Chung et al., 2022), BLOOMZ (1.7B, 3B, 7.1B, 176B-8bit) (Muennighoff et al., 2023) and GPT-4 (OpenAI, 2023). We set the temperatures to zero for all these models to ensure deterministic predictions. We used LLMeBench framework (Dalvi et al., 2024) for the experiments, which provides seamless access to the API endpoints and followed prompting approach reported in (Abdelali et al., 2024).

## 4.5. Prompting Strategy

LLMs produce varied responses depending on the prompt design, which is a complex and iterative process that presents challenges due to the unknown representation of information within different LLMs. The instructions expressed in our prompts include both native (Bangla) and English languages with the input content in Bangla.

### 4.5.1. Zero-shot

We employ zero-shot prompting, providing natural language instructions that describe the task and specify the expected output. This approach enables the LLMs to construct a context that refines the inference space, yielding a more accurate output. In Listing 1, we provide an example of a zero-shot prompt, emphasizing the instructions and placeholders for both input and label. Given that

GPT-4 has the capability to play a role, therefore, we also provide a role for it as an “expert annotator” Along with the instruction we provide the labels to guide the LLMs and provide information on how the LLMs should present their output, aiming to eliminate the need for post-processing.

In our initial set of experiments with BLOOMZ, we observed that it did not respond as effectively to the same instructions as GPT-4. Therefore, we used more straightforward instructions for BLOOMZ, as illustrated in Listing 2. For the other versions of BLOOMZ and Flan-T5, we used the same prompt as BLOOMZ.

```
Instructions:
We would like you to analyze the
    sentiment of the following text.
    Based on the content of the text,
    please classify it as either "
    Positive", "Negative", or "Neutral".
     Provide only the label as your
    response.

text: {input_sample}

label:

role: system,
content: You are an expert annotator.
    Your task is to analyze the text and
     identify sentiment polarity.
```

Listing 1: Zero-shot prompt example for GPT-4.

```
Instructions:
Label the following text as Neutral,
    Positive, or Negative.
Provide only the label as your response.

text: {input_sample}

label:
```

Listing 2: Zero-shot prompt example for BLOOMZ.

### 4.5.2. Few-shot

The seminal work by Brown et al. (2020) demonstrated that few-shot learning offers superior performance when compared to the zero-shot learning setup. This has also been proven by numerous benchmarking studies (e.g., (Ahuja et al., 2023)). In our study, we conducted few-shot experiments using GPT-4 and BLOOMZ. For few-shot learning, we selected examples from the available training data. We used maximal marginal relevance-based (MMR) selection to construct example sets that are deemed relevant and diverse (Carbonell and Goldstein, 1998). This approach has been demonstrated as a successful method for selecting few-shot examples by Ye et al. (2023). The MMR technique calculates the similarity between a test example and the training dataset, subsequently selecting $m$ examples (shots). This selection was performed on top embeddings obtained from multilingual sentence-transformers (Reimers and Gurevych, 2019). We chose to use 3- and 5-shots to optimize the API cost. The effect of $m$ setting will be explored in our future study. Note that our experiments of few-shot with BLOOMZ were worse than zero-shot, which might require further investigation. Therefore, in this study, we do not further discuss the BLOOMZ experiments with few-shot.

In Listing 3, we present an example of a few-shot prompt for GPT-4. The few-shot prompt distinguishes itself from the zero-shot in several ways:

- We provided additional information for the role,
- We simplified the instructions, and
- We included $m$-shot examples.

Our choices of prompts were based on our extensive experiments on similar tasks.

```
Instructions:
Annotate the "text" into "one" of the
    following categories: "Positive", "
    Negative", or "Neutral".
Here are some examples:
Example 1:
text: {input_example}
label: {input_label}

Example 2:
...

text: {input_sample}

label:

role: system,
content: As an AI system, your role is
    to analyze text and classify them as
     'Positive', 'Negative' or 'Neutral'.
     Provide only label and in English.
```

Listing 3: Few-shot prompt example for GPT-4.

## 5. Result and Discussion

In Table 3, we reported the results of our experiments.

### 5.1. Comparison with Baselines:

All experimental setup outperformed random and majority baselines except Flan-T5. We calculate the random baseline by assigning a label to each

| Exp | Acc | P | R | F1 |
|---|---|---|---|---|
| **Baseline** | | | | |
| Random | 33.56 | 38.31 | 33.56 | 33.56 |
| Majority | 49.77 | 24.77 | 49.77 | 49.77 |
| **Classic Models** | | | | |
| SVM | 55.81 | 53.33 | 55.81 | 52.39 |
| RF | 56.75 | 54.61 | 56.75 | 52.62 |
| **Fine-tuning** | | | | |
| Embedding (GPT) | 57.79 | 57.30 | 57.79 | 57.46 |
| Bloomz-560m | 61.71 | 63.08 | 61.97 | 63.08 |
| Bloomz-1.7B | 61.16 | 59.76 | 61.16 | 59.95 |
| BERT-m | 64.95 | 64.92 | 64.95 | 64.90 |
| XLM-r (base) | 66.63 | 66.24 | 66.63 | 66.28 |
| XLM-r (large) | 66.33 | 65.63 | 66.33 | 65.79 |
| BanglaBERT | 69.08 | 67.61 | 69.08 | 67.98 |
| BanglaBERT* | 70.33 | 69.13 | 70.33 | **69.39** |
| **Zero- and Few-shot on LLMs** | | | | |
| **Open Models - 0-shot** | | | | |
| Flan-T5 (large) | 41.28 | 20.23 | 13.77 | 20.23 |
| Flan-T5 (xl) | 49.42 | 29.46 | 18.18 | 29.46 |
| Bloomz-1.7B | 58.33 | 49.38 | 58.33 | 50.38 |
| Bloomz-3B | 59.73 | 50.98 | 59.73 | 51.53 |
| Bloomz-7.1B | 62.83 | 50.92 | 62.83 | 56.24 |
| Bloomz 176B (8bit) | 61.84 | 51.16 | 61.84 | 55.54 |
| Bloomz Majority | 61.97 | 51.32 | 61.97 | 61.97 |
| **Closed Models - $m$-shot** | | | | |
| GPT-4: 0-Shot | 60.21 | 61.65 | 60.21 | 59.99 |
| GPT-4: 0-Shot (BN inst.) | 60.70 | 61.71 | 60.70 | 59.96 |
| GPT-4: 3-Shot | 60.40 | 63.88 | 60.40 | 60.74 |
| GPT-4: 5-Shot | 60.95 | 63.83 | 60.95 | 61.17 |
| GPT-4 Majority | 59.74 | 63.26 | 59.74 | 59.74 |

Table 3: Performance of different sets of experiments. * indicates trained on combined MUBASE, SentiNoB(Islam et al., 2021), and Alam et al. (2021a). BN Ins. refers that instruction is provided in the native Bangla language.

test instance randomly, with the choice of label present in the training set. For the majority baseline, we identify the most common class within the training set and assign this class as the prediction for every instance in the test dataset, subsequently computing the performance.

## 5.2. Performance of Classic Models

The performance of the SVM and Random Forest better than baseline, however, worse than others except Flan-T5. Comparatively, the SVM and Random Forest models exhibit similar performance levels.

## 5.3. Fine-tuning

Fine-tuned models consistently outperform across various settings. Results using GPT embeddings are superior to classical models, though not as effective as some other approaches. Although multilingual models such as BERT-m, XLM-r, and BLOOMZ show promising direction, however, models trained on monolingual text ultimately achieve superior performance.

Given the superior performance of monolingual models across various settings, we chose to augment our training data. By integrating the SentiNoB training set with the MUBASE training set and fine-tuning with BanglaBERT, we managed to boost performance by an additional 1.41% of F1.

When comparing the smaller BLOOMZ model (560m) to the larger one (1.7B), the smaller model performs better. This suggests that more training data might be required to effectively train such a large model. A similar pattern is observed with the XLM-r model when comparing its base and large versions.

## 5.4. Zero- and Few-shot Prompt-Based Results

### 5.4.1. BLOOMZ:

As can be seen in Table 3, the performance of zero- and few-shot approaches is promising, though there is a significant difference compared to the best monolingual fine-tuned transformer-based model. When comparing different parameter sizes of BLOOMZ, we observe that performance increases from 1.7B to 7.1B. However, we see a lower performance with BLOOMZ 176B compared to 7.1B, which might be due to the 8-bit precision.

### 5.4.2. Ensemble:

We hypothesized that predictions from different models might vary, and an ensemble of their outputs might provide better results. Therefore, we opted to use a majority-based ensemble method, resulting in a 5.73% improvement in weighted F1.

### 5.4.3. GPT4:

The performance of GPT-4 is higher than that of other LLMs. Our experiments with different types of prompting did not yield a clear improvement, as can be seen in Table 3. While prior studies on other tasks and languages showed a clear performance gain with a few-shot setup, in our study, we did not find such a gain, only slight differences in precision. Therefore, our future studies will include further investigation of few-shot learning setups.

Our experiments revealed that native language instructions achieved performance comparable to that of English instructions. This indicates the potential for using native language prompts for Bangla sentiment analysys.

While the ensemble of different BLOOMZ settings improved performance, however, it did not help for GPT-4.

#### 5.4.4. Error Analysis on the Output of Prompts:

Further analysis the results of the LLMs outputs we observed that *(i)* Flan-T5 (xl) labeled only five posts as *negative*, and Flan-T5 (large) labeled only 45 posts as *negative*, *(ii)* BLOOMZ completely failed to label posts as *neutral*, and *(iii)* GPT-4 struggled to predict *positive* class.

In Table 4, we present the class-specific classification performance. The results indicate a higher F1 score for the negative class in comparison to the neutral and positive classes. This performance aligns with the class label distribution detailed in Table 2, where ∼50% of the data corresponds to the negative class, followed by ∼31% for the positive class and ∼18% for the neutral class.

| Class | P | R | F1 |
|---|---|---|---|
| Negative | 0.7512 | 0.7771 | 0.7640 |
| Neutral | 0.4616 | 0.3156 | 0.3749 |
| Positive | 0.6871 | 0.7820 | 0.7315 |

Table 4: Detail results on the test set with the model trained using BERT-bn.

## 6. Conclusion

In this study, we present our evaluation of LLMs using zero and few-shot prompting. We offer a detailed comparison with fine-tuned models. Our experiments were conducted on a newly developed dataset named "MUBASE", for which we provide an in-depth analysis. Our results indicate that while LLMs represent a promising research direction, the smaller versions of fine-tuned pre-trained models outperform them. The performance of LLMs suggests that sentiment analysis in a new domain is feasible with reasonable accuracy without the need to develop a new dataset or train a new model. Future research directions include using other recently released datasets and providing a comparative analysis with LLMs. Additionally, further study on few-shot learning represents another promising research avenue.

## Ethics Statement

Our dataset may include posts with negative statements, so we advise model developers to use it cautiously. Using this dataset without careful consideration could lead to misleading results for the audience. Although the dataset's annotation is subjective, we have attempted to minimize this by obtaining annotations from multiple annotators.

Ahmed Abdelali, Hamdy Mubarak, Shammur Chowdhury, Maram Hasanain, Basel Mousi, Sabri Boughorbel, Samir Abdaljalil, Yassine El Kheir, Daniel Izham, Fahim Dalvi, Majd Hawasly, Nizi Nazar, Youssef Elshahawy, Ahmed Ali, Nadir Durrani, Natasa Milic-Frayling, and Firoj Alam. 2024. LAraBench: Benchmarking Arabic AI with large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 487–520, St. Julian's, Malta. Association for Computational Linguistics.

Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Mohamed Ahmed, Kalika Bali, and Sunayana Sitaram. 2023. MEGA: Multilingual evaluation of generative AI. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4232–4267, Singapore. Association for Computational Linguistics.

Firoj Alam, Md. Arid Hasan, Tanvirul Alam, Akib Khan, Janntatul Tajrin, Naira Khan, and Shammur Absar Chowdhury. 2021a. A review of bangla natural language processing tasks and the utility of transformer models. *arXiv preprint arXiv:2107.03844*.

Firoj Alam, Umair Qazi, Muhammad Imran, and Ferda Ofli. 2021b. HumAID: Human-annotated disaster incidents data from twitter with deep learning benchmarks. In *Proceedings of the International AAAI Conference on Web and social media*, volume 15, pages 933–942.

Md Akhter-Uz-Zaman Ashik, Shahriar Shovon, and Summit Haque. 2019. Data set for sentiment analysis on bengali news comments and its baseline evaluation. In *Proc. of ICBSLP*, pages 1–5. IEEE.

Abdullah Aziz Sharfuddin, Md. Nafis Tihami, and Md. Saiful Islam. 2018. A deep recurrent neural network with bilstm model for sentiment classification. In *2018 International Conference on Bangla Speech and Language Processing (ICBSLP)*, pages 1–4. IEEE.

Stephen Bach, Victor Sanh, Zheng Xin Yong, Albert Webson, Colin Raffel, Nihal V Nayak, Abheesht Sharma, Taewoon Kim, M Saiful Bari, Thibault Févry, et al. 2022. Promptsource: An integrated development environment and repository for natural language prompts. In *Proceedings of the*

*60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 93–104.

Nayan Banik and Md Hasan Hafizur Rahman. 2018. Evaluation of naïve bayes and support vector machines on Bangla textual movie reviews. In *Proc. of ICBSLP*, pages 1–6. IEEE.

Francesco Barbieri, Luis Espinosa Anke, and Jose Camacho-Collados. 2022. XLM-T: Multilingual language models in Twitter for sentiment analysis and beyond. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 258–266, Marseille, France. European Language Resources Association.

Vuk Batanović, Boško Nikolić, and Milan Milosavljević. 2016. Reliable baselines for sentiment analysis in resource-limited languages: The serbian movie review dataset. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2688–2696.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Ahmad, Kazi Samin Mubasshir, Md Saiful Islam, Anindya Iqbal, M. Sohel Rahman, and Rifat Shahriyar. 2022. BanglaBERT: Language model pretraining and benchmarks for low-resource language understanding evaluation in Bangla. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1318–1327, Seattle, United States. Association for Computational Linguistics.

Leo Breiman. 2001. Random forests. *Machine learning*, 45(1):5–32.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *Advances in Neural Information Processing Systems*.

Erik Cambria, Qian Liu, Sergio Decherchi, Frank Xing, and Kenneth Kwok. 2022. Senticnet 7: A commonsense-based neurosymbolic ai framework for explainable sentiment analysis. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3829–3839.

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336.

Chenhua Chen, Zhiyang Teng, Zhongqing Wang, and Yue Zhang. 2022. Discrete opinion tree induction for aspect-based sentiment analysis. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2051–2064.

Rumman Rashid Chowdhury, Mohammad Shahadat Hossain, Sazzad Hossain, and Karl Andersson. 2019. Analyzing sentiment of movie reviews in Bangla by applying machine learning techniques. In *Proc. of (ICBSLP)*, pages 1–6. IEEE.

Shaika Chowdhury and Wasifa Chowdhury. 2014. Performing sentiment analysis in Bangla microblog posts. In *2014 International Conference on Informatics, Electronics Vision (ICIEV)*, pages 1–6.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, ACL '20, pages 8440–8451, Online. Association for Computational Linguistics.

Jingfeng Cui, Zhaoxia Wang, Seng-Beng Ho, and Erik Cambria. 2023. Survey on sentiment analysis: evolution of research methods and topics. *Artificial Intelligence Review*, pages 1–42.

Fahim Dalvi, Maram Hasanain, Sabri Boughorbel, Basel Mousi, Samir Abdaljalil, Nizi Nazar, Ahmed Abdelali, Shammur Absar Chowdhury, Hamdy Mubarak, Ahmed Ali, Majd Hawasly, Nadir Durrani, and Firoj Alam. 2024. LLMeBench: A flexible framework for accelerating LLMs benchmarking. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 214–222, St. Julians, Malta. Association for Computational Linguistics.

Kia Dashtipour, Soujanya Poria, Amir Hussain, Erik Cambria, Ahmad YA Hawalah, Alexander Gelbukh, and Qiang Zhou. 2016. Multilingual sentiment analysis: state of the art and independent comparison of techniques. *Cognitive computation*, 8:757–771.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL-HLT '19, pages 4171–4186, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Svitlana Galeshchuk, Ju Qiu, and Julien Jourdan. 2019. Sentiment analysis for multilingual corpora. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, pages 120–125, Florence, Italy. Association for Computational Linguistics.

Md Arid Hasan, Jannatul Tajrin, Shammur Absar Chowdhury, and Firoj Alam. 2020. Sentiment classification in bangla textual content: a comparative study. In *2020 23rd International Conference on Computer and Information Technology (ICCIT)*, pages 1–6. IEEE.

Asif Hassan, Mohammad Rashedul Amin, Abul Kalam Al Azad, and Nabeel Mohammed. 2016. Sentiment analysis on Bangla and romanized Bangla text using deep recurrent models. In *2016 International Workshop on Computational Intelligence (IWCI)*, pages 51–56. IEEE.

Doaa Mohey El-Din Mohamed Hussein. 2018. A survey on sentiment analysis challenges. *Journal of King Saud University-Engineering Sciences*, 30(4):330–338.

Khondoker Ittehadul Islam, Sudipta Kar, Md Saiful Islam, and Mohammad Ruhul Amin. 2021. SentNoB: A dataset for analysing sentiment on noisy bangla texts. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3265–3271.

Md Ekramul Islam, Labib Chowdhury, Faisal Ahamed Khan, Shazzad Hossain, Sourave Hossain, Mohammad Mamun Or Rashid, Nabeel Mohammed, and Mohammad Ruhul Amin. 2023. SentiGOLD: A large bangla gold standard multi-domain sentiment analysis dataset and its evaluation. *arXiv preprint arXiv:2306.06147*.

Md Saiful Islam, Md Ashiqul Islam, Md Afjal Hossain, and Jagoth Jyoti Dey. 2016. Supervised approach of sentimentality extraction from Bengali facebook status. In *2016 19th International Conference on Computer and Information Technology (ICCIT)*, pages 383–387. IEEE.

Mohsinul Kabir, Obayed Bin Mahfuz, Syed Rifat Raiyan, Hasan Mahmud, and Md Kamrul Hasan. 2023. BanglaBook: A large-scale bangla dataset for sentiment analysis from book reviews. *arXiv preprint arXiv:2305.06595*.

Md. Rezaul Karim, Bharathi Raja Chakravarthi, John P. McCrae, and Michael Cochez. 2020. Classification benchmarks for under-resourced Bengali language based on multichannel convolutional-lstm network. *CoRR*, abs / 2004.07807.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Yunlong Liang, Fandong Meng, Jinan Xu, Yufeng Chen, and Jie Zhou. 2022. Msctd: A multimodal sentiment chat translation dataset. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2601–2613.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, et al. 2023. Crosslingual generalization through multitask finetuning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15991–16111.

Shamsuddeen Muhammad, Idris Abdulmumin, Abinew Ayele, Nedjma Ousidhoum, David Adelani, Seid Yimam, Ibrahim Ahmad, Meriem Beloucif, Saif Mohammad, Sebastian Ruder, et al. 2023. Afrisenti: A twitter sentiment analysis benchmark for african languages. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13968–13981.

Md Saddam Hossain Mukta, Md Adnanul Islam, Faisal Ahamed Khan, Afjal Hossain, Shuvanon Razik, Shazzad Hossain, and Jalal Mahmud. 2021. A comprehensive guideline for bengali sentiment annotation. *Transactions on Asian and Low-Resource Language Information Processing*, 21(2):1–19.

Mahmoud Nabil, Mohamed Aly, and Amir Atiya. 2015. Astd: Arabic sentiment tweets dataset. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 2515–2519.

R OpenAI. 2023. Gpt-4 technical report. *arXiv*, pages 2303–08774.

J. Platt. 1998. *Fast Training of Support Vector Machines using Sequential Minimal Optimization*. MIT Press.

A. Rahman and M. S. Hossen. 2019. Sentiment analysis on movie review data using machine learning approach. In *2019 International Conference on Bangla Speech and Language Processing (ICBSLP)*, pages 1–4.

Md. Atikur Rahman and Emon Kumar Dey. 2018. Datasets for aspect-based sentiment analysis in bangla and its baseline evaluation. *Data*, 3(2).

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992.

Sara Rosenthal, Noura Farra, and Preslav Nakov. 2017. SemEval-2017 task 4: Sentiment analysis in Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 502–518, Vancouver, Canada. Association for Computational Linguistics.

Salim Sazzed. 2021. Improving sentiment classification in low-resource bengali language utilizing cross-lingual self-supervised learning. In *International Conference on Applications of Natural Language to Information Systems*, pages 218–230. Springer.

Konstantinos Sechidis, Grigorios Tsoumakas, and Ioannis Vlahavas. 2011. On the stratification of multi-label data. In *Machine Learning and Knowledge Discovery in Databases*, ECML-PKDD '11, pages 145–158, Berlin, Heidelberg. Springer Berlin Heidelberg.

Sadia Sharmin and Danial Chakma. 2021. Attention-based convolutional neural network for bangla sentiment analysis. *Ai & Society*, 36(1):381–396.

Nafis Irtiza Tripto and Mohammed Eunus Ali. 2018. Detecting multilabel sentiment and emotions from Bangla youtube comments. In *2018 International Conference on Bangla Speech and Language Processing (ICBSLP)*, pages 1–6. IEEE.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, EMNLP '20, pages 38–45, Online. Association for Computational Linguistics.

Xi Ye, Srinivasan Iyer, Asli Celikyilmaz, Veselin Stoyanov, Greg Durrett, and Ramakanth Pasunuru. 2023. Complementary explanations for effective in-context learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4469–4484, Toronto, Canada. Association for Computational Linguistics.

Lin Yue, Weitong Chen, Xue Li, Wanli Zuo, and Minghao Yin. 2019. A survey of sentiment analysis in social media. *Knowledge and Information Systems*, 60:617–663.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

## A. Details of the experiments

For the experiments with transformer models, we adhered to the following hyper-parameters during the fine-tuning process. Additionally, we have released all our scripts for the reproducibility.

- Batch size: 8;
- Learning rate (Adam): 2e-5;
- Number of epochs: 10;
- Max seq length: 256.

**Models and Parameters:**

- **BanglaBERT** (csebuetnlp/banglabert):L=12, H=768, A=12, total parameters: 110M; where *L* is the number of layers (i.e., Transformer blocks), *H* is the hidden size, and *A* is the number of self-attention heads; (110M);
- **XLM-RoBERTa** (xlm-roberta-base): L=24, H=1027, A=16; the total number of parameters is 355M.
- **BLOOMZ** (bigscience/bloom-560m): L=24, H=1024, A=16; the total number of parameters is 560M.
- **BLOOMZ** (bigscience/bloom-1b7): L=24, H=2048, A=16; the total number of parameters is 1.7B.