# Detecting Derogatory Comments On Women Using Transformer-Based Models

Fariha Hasan Tonima, Sara Jerin Prithila, Md. Nazrul Islam, Tahsina Tajrim Oishi,
Ehsanur Rahman Rhythm, Adib Muhammad Amit, and Annajiat Alim Rasel
Department of Computer Science and Engineering (CSE)
School of Data and Sciences (SDS)
Brac University
66 Mohakhali, Dhaka - 1212, Bangladesh
{fariha.hasan.tonima, sara.jerin.prithila, tahsina.tajrim.oishi, md.nazrul.islam, ehsanur.rahman.rhythm,
adib.muhammad.amit}@g.bracu.ac.bd, annajiat@gmail.com

***Abstract*—Natural Language Processing(NLP) is a piqued interest field nowadays, as it helps AI to understand and interpret human languages. In order to facilitate the advancement in this field, in this paper, we propose a research on the detection of derogatory comments against women with the help of transformer-based models. This dissertation aims to make a comparative study on how efficient transformer models are in detecting gender biased slandering on languages such as English and Bengali. To carry out this research procedure, the datasets we used were on English and Bengali language which were further trained across the following transformer models: BanglaBERT, XLM-RoBERTa, m-BERT, and DistilBERT.To give further richness to the paper, the Bengali and English datasets used were created by combining multiple different datasets on these languages. The datasets were extracted from various papers related to this or a similar field of research to help reduce biases and to improve its language understanding capability .. Upon, training our datasets across the mentioned models, for the Bangla dataset, Bangla-BERT-Base performed the best with an F1 score of 94% and for the English dataset, m-BERT scored the best with an F1 score of 86.1%. To add on, since the paper mostly focuses on the Bengali language, it will furthermore, encourage others to increase research on low-resourced languages.**

***Index Terms*—Natural Language Processing (NLP), BERT, Transformer, Hate Speech, Bangla Dataset**

## I. Introduction

As we are moving forward into the future, social media is becoming a more intrinsic part of our lives. Well this being said, the rise of its value comes with a few shortcomings as well, and these trade-offs can lead to major risks. Social media platforms from the last decade have shown an intense amount of cybercrimes and bullying, deteriorating health and leading to mental problems for its users; mostly projecting the hate towards women. This is why, it is important to detect such offensive remarks made and stop further slandering.

In this paper, we are going to be using some transformer models: XLM-RoBERTa, BanglaBERT, Bangla-Bert-Base, DistilBERT, and m-BERT; to detect these hateful comments made on social media, but the most highlighted part of our research is that we will be focusing on using a multilingual dataset. The dataset will consist of two different languages: English and Bengali. We aimed to work with Bangla dataset due to heavy research work that has already been done for the English language alone and very few on the Bengali language. These five models are pre-trained and hence can provide a good result on given datasets, and are mostly used due to their ability in analyzing texts and having a solid foundation on contextual and semantic understanding. From the five models mentioned, XLM-RoBERTa and m-BERT are multilingual models which means that the models are already trained in a vast amount of languages. However since multiple languages are detected on these multilingual models the performance of such models on a specific language can be quite low as language bias results due to data scarcity on particular languages, code-switching problems might also arise on mixed languages and errors in translation might also be a huge issue. Thus, fine-tuning the pre-trained models focusing on low-resource languages can reduce the chances of failures mentioned above. To carry out the research, tons of reviews on the latest papers have been done, the dataset has been collected and the methodologies are further elaborated in the upcoming sections.

## II. Literature Review

Over the past few years, a lot of research has been conducted related to this field but the research we aspire to view is a combination of multiple papers published with different innovative approaches to it. To establish a profound insight, in order to overcome our challenges and make a good paper, we have read papers regarding this issue, and the papers are all on the most recent and latest versions of analysis observed in this field.

Rabib et al., in his research, he portrayed the identification of gender discrimination-based cyberbullying in the under-stuffed Bangla language [1]. GenDisc, an archive of Bangla Facebook comments containing gender-based cyberbullying, was created to address this issue. Additionally, a paradigm for detecting cyberbullying is presented in this paper. In their suggested method, four separate models were used to train a gender-based text classifier, which was then applied to the four models using an ensemble strategy. The accuracy of the ensemble predictions was then compared to the accuracy of the individual predictions. Moreover, they used stratified

k-fold cross-validation method to train their dataset. It was also demonstrated that including factors for gender-based discrimination enhances a classifier's ability to distinguish between cyberbullying and other forms of discrimination. Cross-validation tests using the results of our evaluations showed a gender-based speech detection accuracy of 68 percent. In this paper, they have used Max-voting as one of the easiest techniques to combine machine-learning algorithm predictions for classification issues. Max voting predicts and votes for each sample using a base model. The paper also showed us how the ensemble model outperformed the individual models as it had the highest average precision across all folds at 68.07.

Furthermore, Khurana et al. discuss how common internet hate speech is and how automated detection technology has to progress [2]. In social media, people use abusive and offensive words for which they wanted to detect those words. For that, they use the Moj multilingual abusive comment identification dataset to create a unique model that can recognize comments in roughly thirteen distinct languages. In order to deal with hostile behavior, this research develops a unique hate speech detection model that integrates transfer learning and uses state-of-the-art post-processing methods. Moreover, for the bias category comments they used about 300 manually annotated samples. In this research, they used different models as in mBERT and XLM-R etc. In the end, they got an 88.96 percent F1-score test on the Moj Multilingual Abusive Comment Identification dataset.

Afterward, in another paper, Das et al. [3], highlighted the offensive remarks left on social media sites like Twitter and Facebook. This hateful and offensive comments being detected manually is very problematic, hence, automated hate speech and offensive language detection technique is extremely needed. So, they decided to detect those hateful comments using some model. Eventually, there are many works in the English language but low resources in the Bengali language, for this reason, they chose actual Bengali and Roman Bengali Language. After that, they created a standard dataset of 10000 tweets where 5000 were actual Bengali and another 5000 were Roman Bengali. They collected their dataset from Twitter. They used various types of models like m-BERT, XLM-Roberta, MuRIL, and IndicBERT. After that, they note that on their dataset, their model gets a macro-F1-score of 0.754. Additionally, they use both datasets to jointly train the XLM-Roberta model. In conclusion, they got the best result on the MuRIL model after testing their dataset.

Additionally, the paper released by Karim et al., stated that as freedom of expression on social media is increasing it also brings anti-social behaviors like harassment, hate speech, and cyberbullying [4]. Also, there are lots of works on these types of offensive words in English languages but South Asian languages like Bengali have very little work on this. For that reason, they proposed a method they call DeepHateExplainer that is comprehensible for detecting hate speech in the under-resourced Bengali language. Moreover, they worked on about 5000 labeled samples of the Bangla dataset. They work on data by preprocessing and training different types of models like ML Baseline model, DNN baseline models (CNN, Bi-LSTM, and Conv-LSTM), transfer-based model (monolingual Bangla BERT-base, m-BERT, and XLM-RoBERTa). To conclude, their best model was XLM-RoBERTa, which gives an F1-score of 87% and which is (2-5)% better than other models that they trained.

In reference to another paper by Isaksen et al., he portrayed the significance of detecting hateful and offensive language online [5]. In the paper, he also mentioned the challenges of distinguishing hate speech from non-hate speech and offensive language. Offensive language and slurs are not usually used in hate speech to indicate hatred. Here, two datasets including tweets classified as "Hateful," "Normal," or "Offensive" were tested using four deep learners based on the Bidirectional Encoder Representations from Transformers (BERT) with either general or domain-specific language models. The findings show that hate speech is gravely confused with offensive language and normal language by attention-based models. But when it comes to correctly predicting instances of hatred, the pre-trained algorithms exceed cutting-edge findings.

## III. Collected Datasets

### A. For Bangla language datasets

Given the limited availability of gender-based cyberbullying Bangla datasets, we decided to create a dataset specifically focused on the subject of gender-abusive offensive comments. This dataset was compiled by extracting relevant data from other existing datasets. In order to train our models, we have collected different datasets from each source. The majority of the texts in the datasets contain aggressive and profane terms that are prevalent in social media.

For the Bangla dataset, We have compiled textual data from accessible datasets. These datasets contained a vast number of categories of hate speech, including political and religious expressions, sports terms, ridiculing, threats, and other forms of intolerance. However, we focused primarily on the remarks, the majority of which were insults based on gender stereotypes.

The BD-SHS [6] dataset is a manually annotated collection of Bangla Hate Speech. It was derived from publicly accessible material on various platforms such as Facebook, YouTube videos, and Bangla TikTok. During the annotation procedure for binary categorization of hate speech and non-hate speech, each remark underwent annotation by three annotators. From this, we have identified and isolated comments that include obscenities or exhibit an intention to perpetrate harm against someone based on their gender identification.

The Bangla Online Comments Dataset [7] is a collection of comments extracted from Facebook posts posted by actors, singers, and social media influencers. The information has been categorized by experts, revealing that 68.1 percent of the remarks are directed against female victims, including various forms of harassment. Only these comments have been extracted for inclusion in our dataset.

The dataset known as ToxLenbn [8] is a meticulously curated and human-annotated collection of texts that focuses explicitly on the demographic and thematic aspects of toxic language in the Bangla language as seen on various social media platforms. We have collected texts from this dataset that indicate misogynist bullying.

The bnhatespeech [9] dataset comprises a collection of textual data from publicly accessible Bengali articles from several reputable sources in Bangladesh and India. It includes news transcripts from TV channels (ETV Bangla, ZED News), and Bengali articles (Daily Prothom Alo, Dainik Jugasankha). The samples for the hate speech detection dataset were annotated by individuals who are native speakers of Bengali. From this set of data, we have specifically focused on identifying and categorizing comments that exhibit gender-based abuse and extracted those for our dataset.

These diverse datasets exhibited distinct categorizations, including sexual, gender abusive, misogynist bully, gender slander, and so forth. The aforementioned instances have been categorized under the category of 'bully'. To achieve statistical balance, we have used non-hate speech statements that have been categorized as 'non-bully'. Fig 1 shows the dataset has a total of 30,853 texts, with 14,312 texts classified as bully and 16,541 texts classified as non-bully. The dataset we have compiled comprises derogatory and abusive remarks that are specifically targeted toward women.

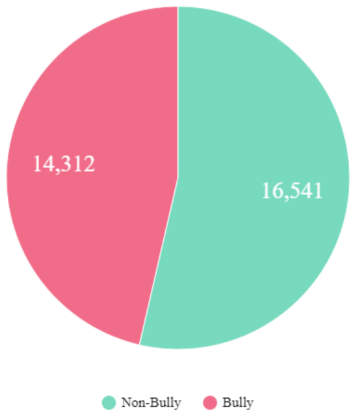


Fig. 1. Class distribution

### B. For English language datasets

An English dataset of 15,138 statements has been created from two other datasets which explicitly indicate gender-based hate content towards women. The first dataset used was on sexist hate speech, this had a corpus of 14,000 data, and the comments in the corpus were labeled either as sexist or not sexist. The huge corpus was collected from different social media platforms like Twitter, Youtube, Instagram, etc. The second dataset [10], the ISEP Sexism dataset, is a small corpus of 1,138 collected statements, the dataset mainly consists of misogynist remarks made towards women in the workplace. The latter dataset contains data that are more hostile and realistic in certain environments than the other. When analyzing the dataset it was found that 55% of the data collected were sexist, the labels were sexist and non-sexist. Moreover, the dataset was already labeled encoded with sexist comments denoting a 1 and non-sexist a 0. This is a publicly available dataset and can be found in kaggle. The corpus built from these two merged datasets is diverse as both are collected from different sources, making it a better benchmark to evaluate the transformer models. Since the new dataset had sexist comments faced from various platforms and environments, it will help the models' understanding of the English language better, in turn making the model more efficient.

## IV. Data Models

### A. BanglaBERT

A popular but low-resource language in the NLP literature, Bangla is used as the pre-trained language for the BanglaBERT Natural Language Understanding (NLU) model, which is BERT-based. Since the majority of downstream task datasets for NLP applications are in the English language, a second model is pre-trained, and since it can learn zero-shot transfer between English and Bangla, it is called BanglaBERT. BanglaBERT outperformed multilingual models and monolingual SahajBERT on all of the tasks in the supervised fine-tuning scenario, earning a BULB score of 77.09% and competing head-to-head with XLM-R (big). In comparison to multilingual and monolingual models, BanglaBERT produces cutting-edge outcomes.

### B. XLM-RoBERTa

XLM stands for cross-lingual Language Model. A multilingual pre-trained model called XLM-RoBERTa (XLM-R) performs better than multilingual BERT. This can be attributed, in part, to the fact that XLM-R was trained using a lot more data. Also trained in 100 languages was XLM-R. XLM Roberta has an extensive vocabulary that includes numerous additional non-English words. Typically, the Python programming language is used to implement this concept in the Transformer library.

### C. mBERT

Multilingual BERT (M-BERT), which may be used with 104 languages, is a single language model that has been pre-trained using monolingual corpora. When mBERT launched, data from all 104 languages was combined. Because of this, mBERT simultaneously comprehends and is aware of the connections between words in all 104 languages. When content from various languages is semantically connected, mBERT recognizes it. M-BERT does provide multilingual representations, but these representations include shortcomings that are specific to certain language pairs and are hence systematic.

### D. DistilBERT

DistilBERT is a BERT-based Transformer model that is compact, quick, affordable, and light. To shrink a BERT model by 40% during the pre-training stage, knowledge distillation

is used. It is a lightweight, quick, and small model that can be utilized for on-device applications and produces acceptable performance results. It is also less expensive to pre-train. It employs a method called distillation that simulates Google's BERT, or giant neural network, by using a smaller one. There are 66 million parameters in the distilled model (DistillBERT), which contains six layers.

## V. Methodologies

### A. Data Preprocessing

Since the obtained corpus is rather noisy, to begin with, the data cleaning process is performed for all datasets, followed by pre-processing before training the collected dataset into the transformer models. The following steps for the method are as follows:

*1) Label encoding:* First, through label encoding, we had to convert the categorical features into numeric features 0 and 1.

*2) Missing and duplicate values:* The dataset was cleaned up by excluding the texts that were missing. the duplicate data were eliminated since it might potentially change the outcome owing to an increase in biases for a certain word.

*3) Bangla special characters removal:* The Bangla text was cleaned up by erasing any unnecessary numerals, special characters, hashtags like Fig 2, and excessive amounts of white space.

৳ ০১২৩৪৫৬৭৮৯ । “ ? ,

Fig. 2. Bangla Special Character

*4) Case Folding:* By using regular expressions we had to convert upper case letters to lower case letters which assisted to maintain consistency flow as well as numeric values were removed using regular expression `[^A-Za-z]` for English dataset.

*5) Different language within Bangla language dataset:* We also had to handle English words in our Bangla dataset too as some of the texts contained Bangla and English languages like Fig 3.

| |
|---|
| এই মহিলাকে দেশের 90% দর্শক dislike করে । যেসব মহিলারা অন্যদের সংসার ধ্বংস করে এদেরকে ইংরেজীতে বলে Home Wreckers.এদের কেউ দেখতে পারেনা . |
| ভাই রে ভাই,সাফা তুই Emergency উমরাহ্ হজ্ব কইরা আয়,যে আকাম করছোছ,ফেজবুকে স্টাটাস দিলে ক্ষমা পাবি না. |
| এই কুফা মহিলার জন্য এখন তোমার ঈদের ছবি নিয়ে struggle করতে হচ্ছে ।এই নাকফাটা মহিলা নিজেকে ঈদের নায়িকা দাবী করত তোমার উপর ভর করে , এতে অন্য qualified নায়িকাদের বদদোয়া লাগছে তোমার উপর । |

Fig. 3. A snapshot of our Bangla dataset

*6) Emoji and white space removal:* Emojis were also replaced with white space. English Punctuation and excessive white spaces were removed using regular expressions and replaced with single white spaces.

*7) Stop word removal:* We applied stop words in Bengali for the Bangla dataset in order to provide greater emphasis on the critical information by deleting terms that are seldom significant from a semantic perspective.

*8) Tokenization:* The text data has to be segmented into distinct tokens before the preprocessing can be carried out. For the DistilBert model, we applied the DistilBert uncased tokenizer, which provides comprehensive tokenization for both punctuation and word pieces using the Transformers library. Additionally, it offers a reduction of 40% in the number of parameters compared to the Bert-base-uncased model. Nevertheless, while training with the Bangla dataset, we needed to raise the maximum length for each of the four separate models, and we also needed to include some special tokenizers.

### B. Data Splitting

Here, the benchmark English dataset required to be split into 2 subsets. The subsets are the training set and the test set. In most cases, the corpus was divided into 0.8 for training and 0.2 for test, furthermore, the way the splitting occurs can either be random or sequential.

For our Bangla dataset to do the train and test splitting, a K-fold split approach was used, with the value of K set to 5. This resulted in the dataset being divided into five successive folds. The Kfold split ensures that every dataset observation is trained and tested equally, preventing over-fitting, and leading to a more unbiased estimation of the model.

### C. Model Training

The pre-trained models have undergone fine-tuning to enhance their performance within the particular context. This method allows us to apply the extensive linguistic knowledge gathered by the pre-trained model to our particular task, which is especially useful for our purpose-specific labeled data.

In this process, a language model learns to anticipate words in phrases, comprehend context, and capture different linguistic properties. The model gains a basic comprehension of language during this phase. The pre-trained model's weights are used as a starting point in the fine-tuning stage, and the model gets access to the task-driven data. To reduce the task-specific impairment function, the parameters of the model are then updated using gradient-based optimization algorithms. Thus, the model can retain the advantages of the enhanced language knowledge it acquired during pre-training while still adjusting its in-built representations to perform well on the new task.

Additionally, fine-tuning has an immense boost in terms of performance. It takes a lot of time and processing power to train a language model from scratch. While fine-tuning makes use of the knowledge that has previously been obtained, it results in more rapid integration and uses fewer resources. In situations where there is a shortage of computational resources or data, this efficiency is very beneficial.

Following the completion of data preparation, we proceeded to the training phase. In this study, binary classification was

conducted using pre-trained models trained on the Hate-Speech-Master-File dataset. As mentioned in data splitting, we divided our dataset into two parts: training and testing. The Training Arguments class from the Transformers library has been employed to establish arguments for a machine-learning model. This library is widely utilized for training and fine-tuning transformers-based models such as BERT. For this, the number of epochs, training batch size, testing batch size, learning rate, warm-up steps, and logging steps have all been set to 3, 8, 10, 2e-6, 32, and 100 respectively. Here, the term "epoch" refers to the frequency with which the model traverses the training dataset throughout the training phase. The selection of an optimal number of epochs is a critical consideration to provide the model with sufficient opportunities to acquire knowledge from the dataset while avoiding the issue of over-fitting.

After that, the batch size parameter dictates the number of training samples handled during each iteration. Then, the learning rate is a crucial parameter that governs the magnitude of steps taken in the parameter space throughout the optimization process. The gradient loss function controls how often the parameters of the model are updated. The identification of an optimal learning rate is of the utmost importance due to the potential consequences of selecting a learning rate that is either too high or excessively low. Specifically, an excessively high learning rate might result in divergence, while an excessively low learning rate can lead to sluggish convergence. The warmup steps refer to the number of initial training steps during which there is a progressive rise in the learning rate. Lastly, the logging stages refer to the process of determining the frequency at which training logs, such as loss values, are shown on the console or recorded in a file. Monitoring the progress of training and assessing the lowering and stabilizing trends of loss are crucial to consider. Moreover, after training, we calculated the accuracy, F1 score, precision, and recall. Here, accuracy is defined as the proportion of correctly categorized events to the overall number of occurrences in the macro F1-score. The macro F1-score, in turn, represents the harmonic mean of recall and precision. Precision is a performance indicator often used in classification tasks to evaluate the effectiveness of a model's optimistic predictions by assessing their accuracy. Finally, to measure the models ability correctly identify instances it needs an actual positive rate, which is known as recall.

TABLE I
HYPER-PARAMETER COMBINATIONS FOR TRAINING BERT VARIANTS.

| Model | Batch size | Learning rate | Epoch | Max length |
|---|---|---|---|---|
| DistillBERT | 8 | 5e-6 | 3 | 2500 |
| XLM-RoBERTa | 8 | 2e-6 | 3 | 5120 |
| BanglaBERT | 8 | 2e-6 | 3 | 5120 |
| m-BERT | 8 | 1e-6 | 3 | 5120 |
| Bangla-Bert-Base | 8 | 1e-6 | 3 | 5120 |

For our model, we have used different sets of epochs, batch sizes, learning rates, warm steps, and logging steps for different models. The best results were found for the different models. The parameters are given in the below table.

### *D. Handling Split datasets for sequence classification of text data:*

After the preprocessing step and data splitting, the output data is fine-tuned in the pre-trained BERT-based models. For these types of models, the preprocessed raw text is firstly assigned with a classification token, [CLS], which is inserted at the beginning of the input sequence. This method is done by the tokenizer. The following steps are taken for the tokenized input sequence:

*1) Embedding Layer:* The input tokenized sequence is passed over to this layer of the transformer so that the tokens get converted into high-dimensional vectors.

*2) Encoder Layer:* Positional encoding (determining the position of the sequence) of the tokenized data is computed in this layer. The layer also consists of two mechanisms. The first one is the multi-head self-attention mechanism where the models learn to capture the contextual information of the embedded input sequence. The next layer is the feed-forward neural network where non-linear transformations take place to capture syntactic information between the input sequence and the relationship between the tokens.

*3) Linear Layer:* In this layer the final hidden state of the [CLS] token, which is the ability of the token to understand the input sequence after it has been passed over from the encoder layer, is mapped from a high-dimensional vector to a vector size equal to the total number of distinct classes present in the classification task.

*4) Softmax Function:* All of the raw output values are converted to class probabilities by this function.

*5) Prediction:* In this final step, the class with the highest probability achieved through the softmax function is provided as the input sequence for determining the final prediction.

From all these layers the model increases its learning rate as it tries to adapt to the task-specific training data and labels, since DistillBERT, XLM-RoBERTa, BanglaBERT, and m-BERT are already pre-trained models, their learning rates will be faster than before. Moreover, improving the models' efficiency in understanding the language, this efficiency occurs due to updating the models' weight function for a better result.

### *E. Model Evaluation Metrics*

This is a very important mechanism in determining whether the model is performing well or badly over the benchmark dataset by looking at the recall, precision, accuracy, and the F1-score for the model.

## VI. RESULT AND ANALYSIS

As seen from the result above, for training the English dataset we have used three of the pre-trained transformer models- DistillBERT, m-BERT, and XLM-RoBERTa, however, we used five models to train the Bangla dataset. This is because the English dataset cannot run on BanglaBERT as it is reserved for only the Bangla language. Moreover, we

have used two BanglaBERT models to see which one provides better performance in training the dataset.

TABLE II
RESULT OF ALL MODELS FOR THE ENGLISH LANGUAGE.

| Models | Precision | Recall | Accuracy | F1-Score |
|---|---|---|---|---|
| DistillBERT | 82.8 | 83.1 | 83.1 | 82.9 |
| m-BERT | 85.9 | 86.4 | 86.4 | **86.1** |
| XLM-RoBERTa | 84.4 | 84.6 | 84.6 | 84.5 |

TABLE III
RESULT OF ALL MODELS FOR THE BANGLA LANGUAGE.

| Models | Precision | Recall | Accuracy | F1-Score |
|---|---|---|---|---|
| DistillBERT | 77.6 | 76.4 | 77.1 | 76.6 |
| m-BERT | 89.0 | 81.0 | 83.0 | 85.0 |
| XLM-RoBERTa | 85.5 | 85.3 | 85.5 | 85.3 |
| Bangla-Bert-Base | 94.0 | 93.0 | 93.0 | **94.0** |
| BanglaBERT | 85.1 | 85.1 | 85.2 | 85.1 |

To deduce the result above for the English dataset, it can be said that m-BERT among the three models, gave the best performance as it has the highest F1-score, 0.861 or 86.1%, from the rest. The F1-score mainly evaluates the model's performance by using the precision and recall values. So, hence we can thus state that the higher the F1-score the better the model. From looking at the evaluation metric table for the Bangla dataset, we can observe that among all the five models run, Bangla-Bert-Base provides the highest F1-score of 0.94 which in terms of percentage is 94%. On top of being the best-performing model for the Bangla dataset, it also gives us the highest score across both the English and Bangla datasets. The next best was provided by XLM-RoBERTa with a score of 0.853 or 85.3% followed by BanglaBERT and m-BERT.

To make a slight comparison with another paper, the result from our dataset on the BanglaBERT model gives a better result than that from the paper DeepHateExplainer [4], the F1-score obtained for BanglaBERT from their dataset was 86% whereas ours made quite a drastic improvement on that, with an increase of 8% from their result. This indicates a great performance improvement for the already existing model leading to more learning and adapting ability of the model on various other datasets.

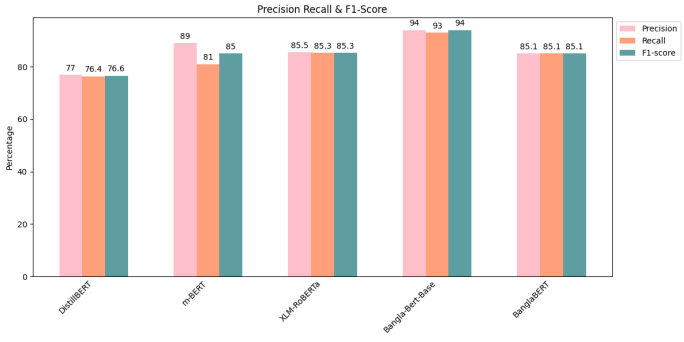


Fig. 4. Comparison of the models for the Bangla dataset.

## VII. CONCLUSION

To conclude, the research work on this paper was mainly focused on a comparative analysis of the transformer model on the combined benchmark dataset. The highlighted part of the paper was on how accurately the model can detect derogatory remarks on women based on low-resource languages like Bangla. Here, we have introduced 2 datasets, one English and one Bangla, which were trained over 5 models. The models we used were DistillBERT, XLM-RoBERTa, m-BERT, and two different BanglaBERT. The results achieved while training for all of the models were pretty decent and among them, Bangla-BERT-base gave the highest F1-score, when trained across the combined Bangla dataset. Transformer models were chosen as a base model instead of other models because of their contextual understanding ability in complex languages. Thus the result established for the Bangla dataset by the Bangla-BERT-base was quite phenomenal with a F1-score of 94%. We hope this paper encourages the training of models on low resourced languages, as hate speech is a growing trend in every country. Furthermore, to add a little, for the upcoming future works, we would like to prepare a hybrid transformer model to train our datasets, and would want to dedicate a paper on a similar topic to determine how the model will work.

## REFERENCES


[1] H. K. Rabib, M. Galib, T. M. Nobo, T. A. Sathi, M. S. Islam, A. R. M. Kamal, and M. A. Hossain, "Gender-based cyberbullying detection for under-resourced bangla language," in *2022 12th International Conference on Electrical and Computer Engineering (ICECE)*. IEEE, 2022, pp. 104–107.

[2] R. Khurana, C. Pandey, P. Gupta, and P. Nagrath, "Animojity: Detecting hate comments in indic languages and analysing bias against content creators," in *Proceedings of the 19th International Conference on Natural Language Processing (ICON)*, 2022, pp. 172–182.

[3] M. Das, S. Banerjee, P. Saha, and A. Mukherjee, "Hate speech and offensive language detection in bengali," *arXiv preprint arXiv:2210.03479*, 2022.

[4] M. R. Karim, S. K. Dey, T. Islam, S. Sarker, M. H. Menon, K. Hossain, M. A. Hossain, and S. Decker, "Deephateexplainer: Explainable hate speech detection in under-resourced bengali language," in *2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 2021, pp. 1–10.

[5] V. Isaksen and B. Gambäck, "Using transfer-based language models to detect hateful and offensive language online," in *Proceedings of the fourth workshop on online abuse and harms*, 2020, pp. 16–27.

[6] N. Romim, M. Ahmed, M. S. Islam, A. S. Sharma, H. Talukder, and M. R. Amin, "Bd-shs: A benchmark dataset for learning to detect online bangla hate speech in different social contexts," *arXiv preprint arXiv:2206.00372*, 2022.

[7] M. F. Ahmed, Z. Mahmud, Z. T. Biash, A. A. N. Ryen, A. Hossain, and F. B. Ashraf, "Bangla text dataset and exploratory analysis for online harassment detection," *arXiv preprint arXiv:2102.02478*, 2021.

[8] M. M. O. Rashid, "Toxlex_bn: A curated dataset of bangla toxic language derived from facebook comment," *Data in Brief*, vol. 43, p. 108416, 2022.

[9] M. R. Karim, B. R. Chakravarti, J. P. McCrae, and M. Cochez, "Classification benchmarks for under-resourced bengali language based on multichannel convolutional-lstm network," in *7th IEEE International Conference on Data Science and Advanced Analytics (IEEE DSAA,2020)*. IEEE, 2020.

[10] D. Grosz and P. Conde-Cespedes, "Automatic Detection of Sexist Statements Commonly Used at the Workplace," in *Pacific Asian Conference on Knowledge Discovery and Data Mining (PAKDD), Wokshop (Learning Data Representation for Clustering) LDRC*, Singapour, Singapore, May 2020. [Online]. Available: https://hal.science/hal-02573576

[11] R. T. Mutanga, N. Naicker, and O. O. Olugbara, "Hate speech detection in twitter using transformer methods," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 9, 2020.

[12] M. F. Ahmed, Z. Mahmud, Z. T. Biash, A. A. N. Ryen, A. Hossain, and F. B. Ashraf, "Cyberbullying detection using deep neural network from social media comments in bangla language," *arXiv preprint arXiv:2106.04506*, 2021.

[13] T. Chavan, O. Gokhale, A. Kane, S. Patankar, and R. Joshi, "My boli: Code-mixed marathi-english corpora, pretrained language models and evaluation benchmarks," *arXiv preprint arXiv:2306.14030*, 2023.

[14] K. Ghosh, D. Sonowal, A. Basumatary, B. Gogoi, and A. Senapati, "Transformer-based hate speech detection in assamese," in *2023 IEEE Guwahati Subsection Conference (GCON)*. IEEE, 2023, pp. 1–5.

[15] E. Guest, B. Vidgen, A. Mittos, N. Sastry, G. Tyson, and H. Margetts, "An expert annotated dataset for the detection of online misogyny," in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, 2021, pp. 1336–1350.

[16] N. S. Samghabadi, P. Patwa, S. Pykl, P. Mukherjee, A. Das, and T. Solorio, "Aggression and misogyny detection using bert: A multi-task approach," in *Proceedings of the second workshop on trolling, aggression and cyberbullying*, 2020, pp. 126–131.

[17] W. Yin and A. Zubiaga, "Towards generalisable hate speech detection: a review on obstacles and solutions," 2021.