2022 3rd Int. Conf. on Innovations in Science, Engineering and Technology (ICISET)
26-27 February 2022, Chittagong, Bangladesh

2022 International Conference on Innovations in Science, Engineering and Technology (ICISET) | 978-1-6654-8397-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICISET54810.2022.9775900

Bengali Hate Speech Detection in Public Facebook Pages

Nasif Istiak Remon
Department of CSE
Metropolitan University
Sylhet, Bangladesh
nasifistiak@gmail.com

Nafisa Hasan Tuli
Department of CSE
Metropolitan University
Sylhet, Bangladesh
nafisahasan98@gmail.com

Ranit Debnath Akash
Department of CSE
Metropolitan University
Sylhet, Bangladesh
ranitid12@gmail.com

***Abstract*—Hate speech is a form of negative communication intended to harm people and communities. Hate speech is quite common in the real world, and it has reached alarming proportions on social media as well. These days our lives have become increasingly reliant on social media platforms, such as Facebook. This is due to the rapid advancement of technology and communication. In Bangladesh, the number of people using social media platforms is also rapidly increasing. In English, detecting hate speech on social media is a difficult task. Comparatively, Bengali is a complicated language with few datasets available. As a result, detection of Bengali hate speech becomes even more challenging. In this paper, we present a new dataset of 10,133 user comments. We have collected them from the comment section of various public Facebook pages. We explore the performance of various machine learning and deep learning models in detecting hate speech. Bengali pre-trained word embeddings from fastText are used to train the models. We are especially interested in Convolutional Neural Network (CNN). To our knowledge it was never used for hate speech detection in binary classification. Another goal of this research is to create a new and large dataset, which will facilitate further research of Bengali Hate Speech Detection. All machine learning and deep learning models performed very well from our experiments. But, Support Vector Machine (SVM) is the one that performed the best among them.**

***Index Terms*—Natural Language Processing (NLP), Bengali Hate Speech Detection, Bengali Text Classification, Word Embedding, fastText, Machine Learning, Deep Learning.**

I. Introduction

People are doing more and more interactions on online social networks. As a result, online hate speech is also on the rise. Research on safety and security in social media has expanded in the last few decades to combat this. But the ever-increasing number of social media sites like Facebook makes everything hard to track. In Bangladesh, many people often face harassment and bullying by strangers. In 2020, rights and legal aid NGO Ain O Salish Kendra (ASK) ran a survey in five districts of Bangladesh. They found that many young students had faced harassment online during the Covid-19 pandemic. Specifically, of the 108 children (61 girls and 47 boys) surveyed, a whopping 30% reported having faced abuse online. At least 56% of them were girls, and 88% had been harassed by strangers [11].

The language used for hate speech is quite different from print media such as newspapers. Bengali hate speech has various linguistic features as well as it is diverse. Hence, research on Bengali hate speech detection is not as rich as it is with English. In this paper, we attempt to improve this situation. We have created a dataset with 10,133 Bengali comments from public Facebook page comment sections. The dataset consists of 3,737 hate and 6,396 non-hate speech. We have collected our dataset from Entertainment, Sports, News, Celebrity, Religious, Atheistic, and Political pages to capture the aforementioned diverse words and linguistic features. We have used several pre-processing steps like - tokenization, stopwords removal, punctuation removal, and stemming for cleaner data. Then we have run several machine learning and deep learning models with word embeddings generated with fastText. [12]. Lastly, we have analyzed our outcome and explained the challenges to detecting hate speech.

II. Background

A. Previous work in Bengali

A new dataset of 30,000 comments is presented in paper [1]. They have conducted baseline experiments with several deep learning models with pre-trained Bengali word embedding such as Word2Vec, fastText, and pre-trained BengFastText. They used YouTube as their primary data source. BengFastText with Long short-term memory (LSTM) and bi-directional LSTM (Bi-LSTM) had the lowest accuracy and F-1 score in their experiment. Deep learning models utilizing BengFastText embeddings overfitted the most, as indicated by their low F-1 score. They surmise BengFastText performed subpar because YouTube data was not used to train it. Their contribution showed the performance of models in BengFastText, and

978-1-6654-8397-1/22/$31.00 ©2022 IEEE

its performance on YouTube data, as BengFastText was not trained on YouTube data prior to their work.

The authors of paper [2] used a variety of ways to find a classifier that was both accurate and optimized for log-loss and hamming-loss. They used SVM and MultinomialNB (MNB), BR with MultinomialNB and GaussianNB (GNB), Classifier Chain and Label Powerset with MultinomialNB, Multilabel k Nearest Neighbours (MLkNN), and Backpropagation for Multilabel Learning (BP-MLL) Neural Networks to create the Binary Relevance (BR) approach. Between their models, BP-MLL has the highest accuracy. Their contribution shows that neural networks perform better than machine learning algorithms.

The authors of paper [3] gathered data from a variety of social media platforms and suggested a root level algorithm for detecting abusive text. They also advised using unigram string features to improve the overall result quality. Their dataset has 300 comments. They stated that it may be difficult to distinguish between hate speech, amusing speech, and abusive speech at times. They surmise their root level algorithm is not up to par with other techniques that exist today, and we believe the dataset is too small and limited in scope for real world use. Regardless, this paper provides insight into algorithm based classification and its pitfalls.

The writers of paper [4] gathered data from a variety of public comment areas on numerous social networks and online resources. The size of their dataset is 4,700. They said they utilized an open-source python module for language identification, but they didn't say what it was called. They employed five Bengali grammar rules in their stemming procedure. The Recurrent Neural Network (RNN) outperforms all other methods. The insight they provide towards the importance of stemming is an important contribution to Bengali NLP research.

Machine learning algorithms are investigated in paper [7] to detect abusive Bengali writing. Facebook comments were gathered for their dataset. They used random texts from the database for their experiment, with 50% of them being toxic and the remainder being non-toxic. SVM achieves the best accuracy for unigram tokens using CountVectorizer vectors. MNB had the best accuracy for bigram tokens using CountVectorizer vectors. They provide a lot of insight into performance between different n-gram levels, and how SVM usually performs best in nearly all classification work in NLP.

B. Previous work in English

The authors of paper [5] created a dataset of 14,509 English tweets. They chose to use supervised classification algorithms to establish lexical baselines. They point out that character n-grams perform well, with 4-grams outperforming all other classes. They introduce an oracle classifier and mention that it has a 91.6% accuracy rate, indicating that none of their features can accurately identify a significant number of their samples. Their work paved the way to detecting hate speech including profanity, but is not hate speech.

The authors of paper [6] collected English tweets containing hate speech keywords using a crowd-sourced hate speech lexicon. They discovered that while racist and homophobic tweets are more likely to be labeled as hate speech, sexist tweets are more likely to be labeled as offensive. They started by stemming words with the Porter stemmer, then built bigram, unigram, and trigram features, each with its own TF-IDF. They discovered that the Logistic Regression (LR) and Linear SVM models outperform other models significantly. One of the most important findings they note is the fact that some slurs and keywords can make or break a prediction, and special care needs to be taken to consider the context in which the words are used.

III. Experimental Setup

A. Dataset preparation

From Facebook, we have collected comments from public pages by using an open-source program called FacePager[1]. After extracting the comments, we have manually checked the whole dataset and removed all the sentences containing non-Bengali language. People do not make comments using the proper Bengali language online as compared to print media. As a result, we need to keep impure text and local dialects. We have removed emoji, punctuation, numerical values, and special characters. In the end, we have created a dataset of 10,133 comments that contain only Bengali sentences. The dataset is available publicly on our GitHub repository.[2]
It is necessary to understand whether a comment contains hate speech or not because hate speech is a subjective matter. We have based our rules by following Facebook community standards[3] on hate speech. They define hate speech as a direct attack against people based on protected characteristics: race, national origin, caste, ethnicity, religious affiliation, disability, sex, sexual orientation, serious diseases, and gender identity. By following these rules, we have come up with our own for our dataset:

- If a person uses comments to attack any person, community, gender, religion, caste, or targets based on their race, gender, physical and mental disability. It will be counted as hate speech. Example:

শালা মাদারচদ
চুদা হবে বেহেস্তে
জানোয়ার মালুর বাচ্চা

- If a comment does not attack anyone and contains no swear words, it is not hate speech. Example:

[1]https://github.com/strohne/Facepager
[2]https://github.com/profake/hate-speech-detection/releases/tag/Rel
[3]https://www.facebook.com/communitystandards/hate_speech

কেমন আছো
সুন্দর লাগছে আপনাকে
ভাইটি রাগ করে চলে গেল

- If a comment contains slang or inappropriate language but does not attack any person, community, gender or anything then it is also not hate speech. Example:

বাল মানে কি
আজ গরিব বলে চটি পড়তে লাগে
তাতে আমার বাল ছেরা গেছে

We have followed these rules, but we have also found a cause for contention here. It is often difficult to differentiate between hate speech and sarcasm. This is due to hate speech being a subjective matter and the Bengali language having various linguistic features. So for this type of comment, we have considered and labeled them as not hate speech. Example:

সময় আসে সময় যায় কিন্তু কিছু মানুষ বোকাচোদাই রয়ে যায়

Pre-processing: After extraction, we have Bengali comments mixed with punctuation, emojis, special characters, and English words. We have followed these steps to pre-process our data:

- Removal of bad characters, punctuation etc: We have removed emoji, punctuation, such as comma(,), semi-colon (;), dash(–), hyphen (–), question mark (?), etc. We have used regular expressions (regex) to remove these characters.
খুব তারাতারি কেয়ামত আসব এদের জন্য। □□□□ -> খুব তারাতারি কেয়ামত আসব এদের জন্য

- Tokenization and stemming: Tokenization is the process of distinguishing between sections of a string of text and optionally categorizing them. We have used the BNLP[4] tool to tokenize Bengali text.
সহমত আপনার সাথে → [সহমত, আপনার, সাথে]

- The technique of reducing inflection in words to their root forms, such as mapping a group of words to the same stem, even if the stem is not a valid word in the language, is known as stemming. We have used bangla-stemmer[5] for stemming.
সহমত আপনার সাথে → সহমত আপন সাথে

- Stopwords removal: Stopwords are words that contribute little meaning to a statement in any language. We have taken them out of every sentence. We have used Stopword ISO[6] for stop words removal.
সে খানে জামায়াত আছে নাকি → খানে জামায়াত

- Word embedding: Word embedding is a form of natural language processing. It is a technique where real-valued vectors represent individual words in a predefined vector space. We have used a pre-trained Bengali word embedding model from fastText[7]. We have prepared word vectors for our entire dataset using this word embedding strategy. We have fed the dataset into our models. These models were trained using Continuous Bag of Words (CBOW) with position-weights, in dimension 300, with character n-grams of length 5, a window of size 5 and 10 negatives.
খানে জামায়াত → [0.05594938 -0.08493394 0.03862553 0.00266161 0.03700361 0.03595718 0.05821856 ...][8]

B. Model development

We have kept 60% of the dataset as training, 20% as development, and 20% as testing set. We have used the following models, fine-tuned them, and kept their best parameters:

- Support Vector Machine (SVM)
  - Kernel = ”rbf” and ”linear”
- Random Forest (RF)
  - Tree count = 81
- Decision Tree (DT)
  - criterion = ’entropy’ and max_depth = 5
- K- Nearest Neighbour (KNN)
  - n_neighbors = 14, metric = ’minkowski’ & p = 2
- Convolutional Neural Network (CNN)
  - kernel_size = 7, loss = binary_crossentropy, activation = ’relu’, optimizer = adam & epoch = 22
- Multilayer Perceptron (MLP)
  - learning rate = 0.002, optimizer = ’adam’, epoch = 8, loss = binary_crossentropy, batch_size = 3 and activation = ’relu’
- Long Short Term Memory (LSTM)
  - epoch = 100, learning rate = 0.01, batch_size = 125, loss = sparse_categorical_crossentropy & units = 64
- Bernoulli Naïve Bayes (BNB) default parameters
- Gaussian Naïve Bayes (GNB) default parameters
- Logistic Regression (LR) default parameters

IV. Results

A. Viewpoints on the result

We have calculated precision, recall, f1-score, and accuracy after implementing and training every one of the models, as shown below:

[4]https://github.com/sagorbrur/bnlp
[5]https://pypi.org/project/bangla-stemmer/
[6]https://github.com/stopwords-iso/stopwords-iso
[7]https://dl.fbaipublicfiles.com/fasttext/vectors-crawl/cc.bn.300.bin.gz
[8]Only 7 dimensions showed. Actual vectors have 300 dimensions.

TABLE I
PERFORMANCE OF DIFFERENT MACHINE LEARNING METHODS

Model	Precision	Recall	F1	Accuracy(%)
SVM (rbf)	0.87	0.87	0.87	87.22
SVM (linear)	0.84	0.84	0.83	83.71
RF	0.83	0.83	0.83	83.02
DT	0.76	0.77	0.76	76.51
KNN	0.83	0.83	0.83	82.93
BNB	0.82	0.81	0.81	81.35
GNB	0.83	0.82	0.83	82.43
LR	0.83	0.83	0.82	82.78

TABLE II
PERFORMANCE OF DIFFERENT DEEP LEARNING METHODS

Model	Precision	Recall	F1	Accuracy(%)
MLP	0.86	0.86	0.86	85.64
CNN	0.85	0.85	0.84	84.70
LSTM	0.77	0.78	0.77	77.79

TABLE III
TRAINING TIME OF ALL METHODS

Model	Training Time (sec)
MLP	96.34
CNN	94.80
LSTM	8304.07
SVM (rbf)	11.65
SVM (linear)	10.53
RF	8.37
DT	2.63
KNN	0.28
BNB	0.05
GNB	0.03
LR	0.17

We have used 7 machine learning methods and achieved around 87.22% accuracy at best in the case of SVM with RBF kernel, as shown in table I. The performance of SVM with linear kernel, RF, KNN, and LR is also good. BNB and GNB both trained quickly and performed admirably. On DT, the training is quick, but it does not perform well.

Table II shows that MLP has outperformed the other deep learning methods with an accuracy of 85.64%. The performance of CNN is likewise comparable, with an accuracy of 84.70% achieved in a similar length of time. LSTM takes the longest time but did not perform well.

B. Why SVM performed the best

As previously mentioned, the Support Vector Machine (SVM) model has outperformed all other machine learning and deep learning models. We have a few clues as to why. Firstly, for moderately sized datasets, SVM with RBF kernel usually performs better than other classifiers and works well for clearly defined classes [9]. Out of all of our models, SVM has the lowest target class overlap. Finally, because our dataset is almost 60-40% hate and non-hate, we can avoid the common pitfall of low performance in an imbalanced dataset [8].

C. Why DT performed the worst

According to [10] the ideal decision-making system may go off track at any point, resulting in poor decisions. They also say that the decision tree's calculation complexity may increase as additional training samples are added. We believe these are the reasons DT has performed the worst out of all the models.

D. Reflecting on the performance of CNN

As mentioned before, we wished to gauge the performance of CNN for text classification. CNN is a neural network generally used for image processing. We have used the 1-dimensional Convolutional Neural Network model, which works on a series of 1-dimensional data compared to the usual 2-dimensional image data. Quite shockingly, CNN is the 3rd best performer out of all our models. It demonstrates CNN's capability in binary text classification.

V. FUTURE PLAN

Hate speech on Facebook is expressed in a very different way than it is in traditional print media. On the internet, there are numerous misspellings, grammatical faults, sarcasm, and other problems. We have also found the same word having multiple spellings but used in expressing the same purpose. For example:

নারি, নারী, নাড়ি, নাড়ী

Though a human brain can comprehend this, a machine will have a tough time doing so. Because some statements are hard to categorize without context, more work in sentiment analysis is required. We have also discovered that people utilize emojis to express themselves, and often they contain the actual meaning of the whole sentence. There is currently no dataset or pre-trained machine that classifies the sentiment of social media emoji. In the future, we would like to work with even more machine learning and deep learning techniques. We will concentrate on increasing the amount of data in this dataset and comparing it to this work in the future. On the same point, we would want to do multivariate categorization to cover all the nuances of hate speech.

VI. CONCLUSION

In this paper, we have implemented various machine learning and deep learning models and compared them. Using the RBF kernel, the Support Vector Machine has fared the best. MLP has the best accuracy in deep learning, while CNN has also done brilliantly. There are some intriguing findings as well. One of which is that some models train quickly and perform well, while some others are both slower in training and bad at performance. Our result analysis shows some pitfalls future researchers can avoid and the importance of sentiment analysis while working on online mediums. Our publicly available dataset can also be used by future researchers to base their work on, or to speed it up. We expect that the trained models will make online places safer for all Bengali speakers and that our work will motivate further research on this topic.

REFERENCES

[1] N. Romim, M. Ahmed, H. Talukder and Md S. Islam, "Hate speech detection in the Bengali language: a dataset and its baseline evaluation," International Joint Conference on Advances in Computational Intelligence, pp. 457–468, November 2020.

[2] A.N.M. JuBaer, A. Sayem and Md. A. Rahman, "Bangla toxic comment classification (machine learning and deep learning approach)," 8th International Conference System Modeling and Advancement in Research Trends, pp. 62–66, November 2019.

[3] Md G. Hussain, T. Al Mahmud and W. Akthar, "An approach to detect abusive Bangla text," International Conference on Innovation in Engineering and Technology (ICIET), December 2018.

[4] E. A. Emon, S. Rahman, J. Banarjee, A. K. Das and T. Mittra, "A deep learning approach to detect abusive Bengali text," 7th International Conference on Smart Computing & Communications, June 2019.

[5] S. Malmasi and M. Zampieri, "Detecting hate speech in social media," Proceedings of the International Conference Recent Advances in Natural Language Processing, September 2017.

[6] T. Davidson, D. Warmsley, M. Macy and I. Weber, "Automated hate speech detection and the problem of offensive language," The 11th International Conference on Web and Social Media, 2017.

[7] S. C. Eshan and Md S. Hasan, "An application of machine learning to detect abusive Bengali text," International Conference on Computer and Information Technology, December 2017.

[8] R. Akbani, S. Kwek, and N. Japkowicz, "Applying support vector machines to imbalanced datasets", 15th European Conference on Machine Learning, pp. 39–50, September 2004.

[9] I. Ahmad, M. Basheri, M. J. Iqbal and A. Rahim, "Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection," IEEE Access, vol. 6, May 2018.

[10] B. T. Jijo and A. M. Abdulazeez, "Classification based on decision tree algorithm for machine learning," Journal of Applied Science and Technology Trends, 2021.

[11] A. Jahin, "The real and intangible threat of online child harassment," The Daily Star, March 2021.

[12] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, "Enriching word vectors with subword information," Transactions of the Association for Computational Linguistics, vol. 5, pp. 135–146, June 2017.