Evaluating Gender Bias in Textual Data Using NBias: A Natural Language Processing Perspective

*

Protiva Das
Department of CSE
BRAC University
Dhaka, Bangladesh
protva.das@g.bracu.ac.bd

Annajiat Alim Rasel
Department of CSE
BRAC University
Dhaka, Bangladesh
annajiat@gmai

Tamima Binte Wahab
Department of CSE
BRAC University
Dhaka, Bangladesh
tamimabw@gmail.com

***Abstract*—This study tackles the challenge of detecting gender bias in textual comments, a crucial issue given the global concern over gender-related discrimination. Utilizing a secondary dataset from Kaggle, we explored the effectiveness of various natural language processing models, including LSTM, Bi-LSTM, GRU, Bi-GRU, BERT, and DistilBERT, to identify and analyze gender biases. Our approach began with preprocessing the data, followed by splitting it into training and test sets and then applying the models to determine which best detects biases at an early stage. The findings reveal that certain models are more adept at highlighting subtle biases, contributing significantly to the efforts of making online and offline communication spaces more inclusive. Looking forward, this research opens avenues for addressing other forms of bias such as political or religious, thereby broadening the impact of our work on society.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

In today's world, what we say online can shape how people see and treat each other. Unfortunately, comments on the internet often show bias against certain genders, which can lead to unfair stereotypes and hurt feelings. This kind of bias isn't just mean; it can also make people feel unsafe or unwelcome. Given how big these issues have become globally, it's more important than ever to address them. Our research aims to spot these gender biases in in-text comments. We decided to tackle this because we want to help reduce the harm caused by biased comments and make online spaces nicer for everyone. While there has been plenty of work on spotting hate speech, our project is different because it looks specifically at how biases against all genders show up in what people write, not just on social platforms but anywhere. To do this, we got many comments from a dataset on Kaggle, a website where people share data. We cleaned up this data and organized it, then used some smart tech tools like LSTM, Bi-LSTM, GRU, Bi-GRU, BERT, and DistilBERT to see which could best find the bias. Our goal is to catch these biases early on so we can do something about them and make the internet a better place for talking and sharing ideas.

Identify applicable funding agency here. If none, delete this.

II. Literature Review

These studies utilizing transformer-based token classification models have demonstrated capability in detecting biased language across various domains. However, these models struggle with misspellings, changes in severity, embedded biases in training data, and performance inconsistencies across languages and cultural contexts. These limitations of [1] underscore the need for refined approaches to enhance accuracy and generalizability in bias detection.

Evaluated bias detection in narratives generated by large language models like GPT-3.5, using it both to produce and analyze stories for [2] inherent biases, and employing a fine-tuned GPT-2 for dedicated bias classification. Despite achieving a high accuracy of 97.5

[3] Introduces the Contextualized Bi-Directional Dual Transformer (CBDT) Classifier, employing two synergistic transformers for enhanced bias detection. However, its focus remains on English textual data, omitting biases in images and audio and not incorporating newer technologies like LLAMA 2 and GPT-4. These limitations highlight opportunities for broader and technologically updated investigations.

Researchers worked on the [4] Contextual-Dual Bias Reduction Recommendation System (C-DBRRS) for news, tested on two real-world datasets. Challenges include limited scalability due to computational complexity, sensitivity to hyperparameter settings, and potentially inaccurate assumptions about the importance of temporal dynamics in user-item interactions. These limitations suggest critical areas for further model optimization and broader applicability testing.

Multilabel classification of toxic comments using word embeddings (GloVe, Word2vec, FastText) and deep neural networks (NN, CNN, RNN, LSTM, GRU), with the BiGRU model achieving the highest performance metrics were worked on [5]. Despite these successes, the research highlighted a need for improved strategies to address imbalanced data effectively.

Introduced an LSTM-LM transfer learning model designed to automatically identify bias in enterprise content using manually tagged documents, achieving an accuracy of 0.89. It [6]

compared this approach against 10 baseline models, effectively demonstrating the LSTM-LM's robustness in performance. This underscores the model's potential for enhancing bias detection processes within organizational settings. Headline Attention Network for bias detection in news articles, uniquely designed to mimic human reading patterns and incorporate an attention mechanism focused on headlines. This model in [7] achieved 89.54

The study [8] introduced DA-RoBERTa, DA-BERT, and DA-BART for sentence-level media bias detection, achieving an F1 score of 0.814. While effective, the models are limited to sentence-level bias and evaluations on a single dataset. Future improvements should extend capabilities to detect biases across different textual levels and develop more nuanced evaluation metrics to enhance detection accuracy comprehensively.

BERT-BiGRU model, combining BERT's contextual word representations with BiGRU's bidirectional text feature extraction for text classification in [9]. Achieving high metrics (accuracy, recall, F1 score all above 0.9), the model is however resource-intensive due to BERT's complexity, relying on Google's pre-trained models. Optimization is needed to reduce resource consumption and training time.

The research project examined LSTM and GRU models and used a Bi-GRU-LSTM-CNN classifier for deep learning, with an F1-score of 70.576

For stance identification, a two-channel CNN-GRU fusion network outperformed SVM, CNN, GRU, and an earlier hybrid model in [11] and got accuracy and F1 scores. To improve the method and use it for public opinion analysis, this network includes convolutional layers for local feature extraction and GRU for time characteristics. In the future, it is intended to add an attention mechanism and expand the dataset.

The study in [12] presents a bidirectional LSTM model for fake news detection, demonstrating superior accuracy over CNN, RNN, and unidirectional LSTM using two unstructured news datasets. It achieved a 4.18

A. Methodologies

At first, the data are gathered properly for performing bias detection from numerous social media comments. Most of the comments are sexism-based. The comments are subcategorized into multiple factors. For, evaluating the model properly, the model has been split into three categories namely, training, testing and validation. At first, the model is trained on 70% data from the gathered dataset. Later, the remaining 30% data are preserved for the validation set. Finally, the test data is supplied from outside comments. The dataset needs to be annotated properly in order to understand the model precisely. More than, 20,000 comments are available in this dataset where there are multiple columns representing different kinds of biasness towards people.The main categories of the comments are:

1) **ID Rewire**
2) **Comment**
3) **Sexiest label**
4) **Category label**
5) **Vetor label**

Here, Rewire id represents two things. The first thins is the year in which the comment was made. Secondly, what type of person has made the comment. The biased comment related to sexism denote what the actual comment is. After that, the sexiest label category represents whether the comment is sexiest or not. Be specificly, what type of sexiest comment this is. In the dataset, there is also a sub sector that is known as label vector. The subcategory of the comment is denoted by the Label vector. All the necessary attributes are available inside this dataset. That is the cardinal reason for selecting this dataset.

1) **Animosity**
2) **Online threats**
3) **SPrejudiced Discussion**
4) **Derogation**

From data exploration, we can see that there are 20,000 comments in total. Twelve thousand are related to sexism, and others are not related to sexism. The comments subcategories are also important for processing data.

B. Preprocessing of the dataset

Data preprocessing is vital for providing data to numerous architectures to understand semantics precisely. It includes a good number of operations, which depend on the language and pattern of the dataset. Another important part is ensuring the model is not overfitted because of the anomaly in the dataset. For Bengali, numerous preprocessing techniques are available. Hence, the authors have used only the vital operations required to provide the DL and ML architectures. The adopted preprocessing techniques are stated below:

1) **Dropping Null Values:** The very thing is to drop the null values from the dataset. We have utilized the Pyhton nltk library to operate. Other than that, all other columns are eradicated from the dataset except for the comment attribute.
2) **Removing Stopwords and Keywords:** Bengali language consists of many stopwords that do not provide any special meaning to the models. On the contrary, they introduce ambiguity while capturing the inner meaning of these sentences. That is why all the stopwords have been removed. Another vital factor is to eradicate the punctuation marks from the sentences. Punctuation marks also do not provide any special meaning in these sectors.
3) **Special Character Removal:** All the special characters from the comments have been removed. To fulfill this purpose, the RE module of Python is used.
4) **Tokenization:** Tokenization is the process of dividing a written document into smaller pieces so that machine learning methods can be used. We have converted all the big comments into smaller tokens. That allows the models to understand and capture the words that are helpful to understand the semantics. The Bangla BERT

tokenizer has been utilized to provide the model to the Bangla BERT model.

5) **Creating Dictionary:** A dictionary has been created to identify unique words. The word definition is also created using this dictionary. As we are mining semantics from the texts, we have created a dictionary to extract the inside information with precision. The word confusion can be resolved with the aid of this dictionary. Figure **??** represents the most occurred words encountered approximately 309 times. Apart from this, there are 7404 words available in the dataset, where 507 words are unique. After closely understanding the words, it can be observed that most of the words are related to finance.
6) **Stemming:** After observing closely, the third and fourth words disclose the same meaning but differ in spelling. The primary purpose of stemming is to convert the data into its base form. As a result, it becomes easier for the models to understand the meaning. As a result, the number of variations reduces to a greater number. In this research, stemming also plays a vital part in handling the sparsity of the feature space.
7) **Word Embedding:** The cardinal purpose of using Word Embedding is to represent data in a dense vector in a vector space. Word embedding is a proper solution for a one-hot encoder that produces high-dimensional vectors. Learning distributed representations of words based on their context in a sizable corpus of text data is the fundamental notion behind word embedding. In this research, authors have utilized neural network word embedding techniques such as Word2Vec and Glove vectorizer. For the Bangla BERT model, no embedding technique has been adopted as this model incorporates word embedding into its architecture.