Gender-Based Hate Comment Detection with NLP

by

Aaraf Bin Rashid
18101010

CSE424

Supervisor: Annajiat Alim Rasel
Date of Submission: 25 April 2024

Department of Computer Science and Engineering
Brac University

© 2024. Brac University
All rights reserved.

Abstract

The widespread accessibility of social media undoubtedly has its perks but the misuse makes it harder to believe in its blessings. Hate comments on different social media platforms have become a continuous issue that reflects social inequity and moral decay. To overcome this issue, we came up with the idea of using Natural Language Processing (NLP) techniques such as Recurrent Neural Networks (RNNs) that give better results at detecting hate comments based on gender. The goal is to develop a robust model that will be trained on a dataset combined with sexist and non-sexist language. By detecting sexism we intend to create a safer space for individuals over the different platforms of the internet. We will use the tokenizer API from Tensorflow Keras, we will transform text data into numeric sequences. Our model incorporates a Sequential architecture with an embedding layer mapping words to N-dimensional vectors, a GlobalAveragePooling1D layer for dimensionality reduction, and a dense classifier.

Keywords: Neural Network, LSTM, Text Classification, Sexism

Table of Contents

Chapter 1

Introduction

1.1 Introduction

The surge in popularity of internet platforms has made social media more accessible to the general public. While this level of accessibility has made our lives easier, it has also resulted in growing misuse of these sites. One such example is the ongoing problem of gender-based hate comments, which persists throughout our online lives. Gender-based hate comments reflect a culture of inequity and injustice, demonstrating the moral decay of the human race.

To address this issue square on, we devised the use of Natural Language Processing (NLP) to detect such gender-based hate remarks. In this study, we propose the use of Recurrent Neural Networks (RNNs), which are designed to accept sequential data and then perform text categorization, resulting in the detection of gender-based hate remarks on online platforms.

The primary goal of our team is to develop a robust system comprised of many sophisticated models that will be trained on a dataset including sexist and non-sexist language. The technology will evaluate the data and then classify the remarks as sexist or non-sexist. We hope that by detecting hate remarks on social media platforms, we can make the internet a safer and more peaceful place for individuals of all backgrounds.

Chapter 2

Literature Review

The study 'Detection and Classification of Sexism on Social Media Using Multiple Languages, Transformers, and Ensemble Models' discusses detecting and categorizing sexist content in social media posts. For this study, the researchers used well-known transformer topologies such as BERT, Roberta, Electra, GPT2, XLM-Align, and Info-XLM. Their research methodology is based on two critical tasks: detecting and categorizing sexism. The study relied on textual data in English and Spanish that included both single-language and bilingual versions. Data collection, data processing, exploratory data analysis, model implementation and hyperparameters analysis, final model implementation, and model comparison are the six systematic processes in this research technique. The models were trained using the dataset EXIST 2022 shared task, which included 11345 labeled posts from Twitter and Gab in English and Spanish, and then evaluated using 1058 tweets from January 2022. The accuracy and f1-Macro metrics were used to measure the model's performance, while Python was employed as the programming language. The models BERT and RoBERTa performed far better than the other designs, and multilingual models outperformed single-language models. RoBERTa received an F1 macro score of 0.954 in English, while Info-XLM had a score of 0.830 in Spanish.

In the paper 'Hate Speech and Counter Speech Detection: Conversational Context Does Matter' the researchers worked on the conversational context in identifying hate speech and counter speech. Hate speech is expressed in a way that degrades any particular individual or group or provokes them to become violent. It often leads to violence and inequity. As the spread of hate speech is increasing day by day, it has become important to take precautions. Two approaches like disruption and counter speech can be effective in this regard. The authors have given a dataset of Reddit comments annotated with hate, neutral, or counter-hate labels that showcased the impact of context on human judgments. Then again, experiments with context-aware classifiers gives us evident improvements in detecting hate speech and counter speech. In his paper a comprehensive analysis was done on language patterns and the benefits of including context in hate speech detection. A neural network model that had a pretrained RoBERTa transformer, that had a fully connected layer for classification purposes. Two textual inputs are taken into consideration: the Target alone and the Parent and Target together. The paper uses a dataset consisting of Gold instances. The dataset was randomly split into training (70%), validation (15%), and testing (15%) sets. Subsequently for training, silver

instances are used only. The authors did experiment with two strategies to evaluate performance: blending Gold and Silver annotations and pretraining the models with related tasks. The results prove that if blending Gold and Silver annotations are used, the performance gets better compared to using only Gold instances. The percentage of F1 weighted average is 61 % and 58 % consequently. If pertaining is done using stance detection data, the result of F1 improves. Incorporating the Parent comment in the training results into higher F1 scores for all classes and a higher weighted average (F1 weighted average: 0.63 for models trained on both Parent and Target, compared to 0.58 for Target-only models). The best performance is achieved by blending Gold and Silver annotations and pretraining with stance detection (F1 weighted average: 0.64). The paper surely has limitations such as considering the parent comment as context and potential biases through keyword sampling. Future works can be done to explore additional context strategies and present biases through community-based sampling.
The paper 'CoRoSeOf - An Annotated Corpus of Romanian Sexist and Offensive Tweets' introduces CoRoSeOf, a large corpus of Romanian social media manually annotated for sexist and offensive language. Detecting sexism is extremely important because it allows us to promote fairness and equality, create spaces where everyone feels safe and respected, prevent harm caused by gender-based discrimination, increase awareness about the issue, hold individuals and institutions responsible for their actions.. The purpose of our study is to address and prevent sexist language online. In this paper the aim was to create a Romanian corpus annotated specifically for sexist language on Twitter. By recognizing and detecting sexist language, users can be protected and fostering more inclusive digital spaces will be possible. To achieve the goals, authors identified the need for a publicly available corpus for classification tasks. They introduce CoRoSeOf, a large Romanian corpus manually annotated for sexist language on Twitter. A team of ten annotators, consisting of seven females and three males, carried out the annotation task. They were all native speakers of Romanian and university students majoring in Languages and Literature or Modern Applied Languages. Before starting the annotation task, the annotators underwent a training period. To annotate the 40,000 tweets, Google Forms was used. The data was randomly split into 160 forms, each containing 250 tweets, and each form was assigned to three annotators. At the end of the annotation process, each tweet received three annotations. The binary classification results on the CoRoSeOf corpus show an accuracy of 83.14% (F1 score), with precision and recall at 83.07% and 83.24% respectively. These scores surpass the best results achieved on the French sexist language corpus. False positives for offensive tweets are 4.69%, while false negatives occur due to the absence of explicit sexist keywords. The classifier aligns better with unanimous annotator agreement, resulting in fewer mistakes. Three-way classification into sexist language types also performs well, with macro-averaged precision, recall, and F1 scores at 71.62%, 69.29%, and 70.02% respectively. The dataset shows consistent annotations and will benefit researchers and NLP system development. Future work involves further experiments, improvements, and exploration of different forms of sexism. Data collection will expand to capture a broader range of sexist and offensive language.

In this paper "Bengali Slang detection using state-of-the-art supervised models from a given text", supervised machine learning models are used to identify Bengali slang phrases in texts from social media. A corpus of 5000 utterances in Bengali, labeled with slang and non-slang terms, is compiled and annotated by the authors. The seven supervised models that they examine are Naive Bayes, Logistic Regression, Support Vector Machine, Decision Tree, Random Forest, K-Nearest Neighbor, and Artificial Neural Network. With the help of metrics like accuracy, precision, recall, and F1 scores, they assess the models. They discover that, with 70% accuracy and 86% recall, the artificial neural network model works best. They propose that their systems can be connected with social media sites to limit the usage of Bengali slang in online.They also point out some of the limitations of their research, including the small corpus size, the lack of regional variation in the slang terms, and the fluidity of the slang language. To increase the performance of the models, they suggest using deep learning techniques, increasing the corpus with more slang phrases from various locations, and creating a slang translator for Bengali.

Chapter 3

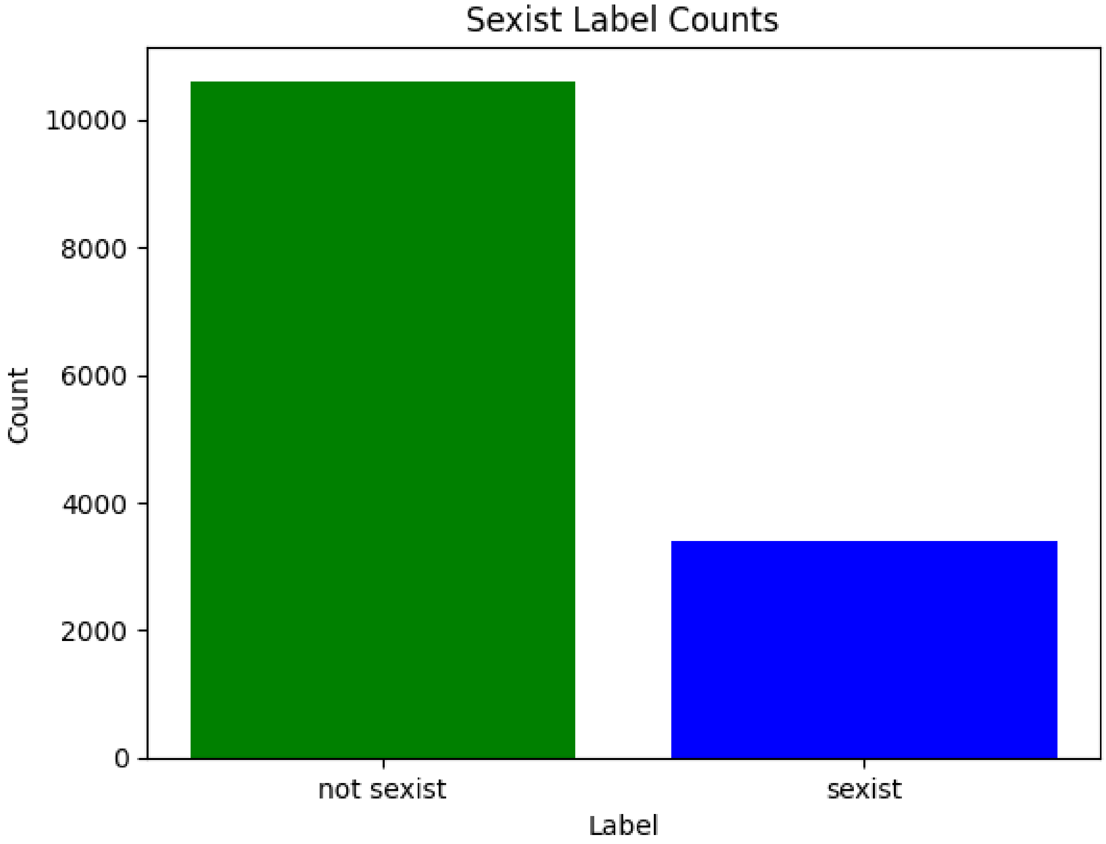
Methodologies

In this study, we aimed to develop a machine learning model to detect sexism by using Recurrent Neural Networks (RNNs). Firstly, we preprocessed the dataset by dropping unnecessary columns and visualizing the data to identify any imbalances, which we corrected by using under sampling. Then, we converted the labeled data into numeric values, where we set sexist as 1 and not sexist as 0. The dataset was then divided into three parts, namely train, test, and dev sets. To convert the text data into numerical data, we utilized the Tokenizer API from Tensorflow Keras and represented each sentence by sequences of numbers using texts to sequences. We then padded the sequences to ensure each sequence had the same length. We used the Sequential calls for Keras sequential model, in which the embedding layer mapped each word to an N-dimensional vector of real numbers, with the embedding dimension set at 16. As the embedding layer was the first hidden layer in our model, we set our input layer by defining the input length as maximum length. We then used GlobalAveragePooling1D as the pooling layer, which helped to reduce the number of parameters in the model. We defined the dense classifier, compiled it, and trained the model. For model selection, we chose LSTM and Bi-LSTM models and trained them using the same train-test-split data with an 80-20 ratio. The evaluation was based on the test data, using a batch size of 256 and epoch value of 30.

Chapter 4

Dataset

The dataset contains 14000 texts with labels.There are some id which is an unique identifier for each comment in the dataset. Then there is the actual text of the comments. These are the statements or sentences that should be analyzed to determine whether they are sexist or non-sexist. There are also labels of the comments which are either sexist and not sexist.These labels are used as the target variable in a classification task, where the goal is to predict whether a given comment is sexist or not.There are also category of the comment.There is no null data in the dataset. Around 25% of the data is sexist and 75% of the data is not sexist.



Chapter 5

Model Implementation:

5.0.1 Text Tokenization and Padding:

We determined the parameters like maximum sequence length, truncation and padding types, out-of-vocabulary token and vocabulary size after splitting the dataset. We used a tokenizer to fit the training and dev data and index the words. Then the text data is converted to a sequence by using the tokenizer and pad the sequences to a compatible length.

5.0.2 Model Architecture:

We used various model architectures for text classification through sequential models taken from TensorFlow/Keras

Model 1: This is a model that has an embedding layer, global average pooling, a dense layer, dropout, and a final dense layer with a sigmoid activation.

Model 2: A Bidirectional LSTM-based model that includes a embedding layer, Bidirectional LSTM layer, dropout, and a final dense layer with a sigmoid activation.We compile it with binary cross-entropy loss, the Adam optimizer, and accuracy as a metric.The fit function trained the models on the training data, specified the number of epochs, validation data, and early stopping callback.

Chapter 6

Results

The validation loss for the Dense, LSTM and Bi-LSTM models are 0.5915, 0.6717 and 0.5795 respectively. The validation accuracies are 68.46%, 60.59 %, and 71.32%. The F1 scores are 68.05%, 65.33% and 68.85%. Based on the loss, accuracy and the plots, we can conclude that the Bi-LSTM model is the best model for this classification case. We tried testing the model with two sentences, one is a sexist comment and the other one is not sexist. We tried to evaluate using Bi-LSTMmodel and it predicted correctly.

Future Work

This study opens up numerous possibilities for future exploration and improvement. The following regions can be investigated:
Contextual Insight: Improving current models to better grasp conversation circumstances. Multilingual Adaptation: Extending existing models to handle many languages for global and diversified applications. Optimized parameters: Model settings that have been fine-tuned for better and more accurate performance. Evaluation of Robustness: Ensuring that the models can withstand real-world attacks. Advanced Architectures: Investigating similar transformer concepts such as GPT-3, BERT, and RoBERTa. Human Touch: Incorporating user feedback to improve the system.

Conclusion

To summarize, this study applies Natural Language Processing approaches to address the persistent issue of gender-based hate comments on social media sites. The construction and evaluation of the recurrent neural networks used, particularly the Bi-LSTM model, demonstrates the effective classification of sexist and non-sexist remarks discovered in internet sources. The system's performance illustrates Natural Language Processing's promise for building a safer and more open online environment for everyone. However, the problem is evolving, and the models described in this study are an attempt to counteract it and eliminate the new strategies of sexist hate remarks. This project serves as a stepping stone toward the development of a sophisticated and resilient system that allows us to envision a brighter future with a more harmonious digital world for all individuals.

@miscrani$_{2}$021$_{n}lp, author = Rani, Vijaya, month = 04, title = NLPTutorialforTextClassificatio$
$https : //medium.com/analytics - vidhya/nlp - tutorial - for - text - classification - in - pyt$
2021$, organization = Medium$
@miscpython$_{p}ractical, author = Python, Real, title = PracticalTextClassificationWithPythonan$
$https : //realpython.com/python - keras - text - classification/, organization =$
$realpython.com$