Advancing Online Hate Speech Detection Using External Features and Large Language Models

by

Amit Das

A dissertation submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Auburn, Alabama
August 3, 2024

Keywords: Hate Speech, User Gender, Large Language Model, Natural Language Processing

Approved by

Cheryl D. Seals, Chair, Professor of Computer science and Software Engineering
Gerry V. Dozier, Professor of Computer Science and Software Engineering
Jakita O. Thomas, Associate Professor of Computer Science and Software Engineering
Mary J. Sandage, Professor of Speech, Language & Hearing Sciences
Lauramarie Pope, Assistant Professor of Speech, Language & Hearing Sciences
Myoung-Gi Chon, Associate Professor of Communication & Journalism

Abstract

Social media is a concept developed to link people and make the globe smaller. But it has recently developed into a center for hateful posts that target different people and communities. As a result, there are more events of hostile actions and harassing remarks present online. Since this issue can cause immense harm on a person, it needs to be addressed with immense priority.

There are many Natural Language Processing models that have been implemented for hate speech detection. In our study, we begin by using BERT combined with TFIDF representation to tackle the challenges of identifying irony and stereotype-spreading authors on Twitter. For the classification task, we employed a logistic regression classifier. Our findings indicated that the combination of BERT representation with TFIDF yielded very promising results.

To delve deeper into the issue, we addressed sexism, another form of hate speech that predominantly targets women. For sexism detection, we introduced a fine-tuned RoBERTa model. This involved encoding the initial data representation using RoBERTa and implementing three distinct Multilayer Perceptrons (MLPs) for the three sub-tasks. The experimental results showcased the effectiveness of our proposed strategy.

Additionally, we explored the potential benefits of incorporating external features in the detection of sexism and hate speech. Specifically, we examined the impact of user gender information on online sexism detection in both binary and multi-class classification contexts. Given that most sexist comments are directed towards individuals of a particular gender, understanding the role of user gender information is crucial. Our experiments demonstrated that integrating user gender information with textual features enhanced classification performance in both binary and multi-class classifications.

Further advancing our research, we introduced OffensiveLang, a novel community-based implicit offensive language dataset generated by ChatGPT 3.5, covering 38 different target

groups. Despite ethical constraints limiting the generation of offensive texts via ChatGPT, we devised a prompt-based approach to effectively generate implicit offensive language. To ensure data quality, we evaluated our dataset through human assessments. Moreover, we employed a prompt-based Zero-Shot method with ChatGPT and compared detection results between human annotations and ChatGPT annotations. We also utilized existing state-of-the-art models to evaluate their effectiveness in detecting such languages and investigated annotator biases in hate speech data annotation using large language models.

Lastly, we investigated gender, race, religion, and disability biases in LLMs used for hate speech detection and proposed mitigation strategies. We demonstrated the presence of these biases in LLMs such as GPT-4o and GPT-3.5 when annotating hate speech data. We then explored the underlying factors that contribute to these biases, providing a thorough analysis of the annotated data and emphasizing the role of subjective interpretations. Finally, we suggested potential solutions to mitigate these biases, highlighting the importance of tailored prompts and fine-tuning LLMs to enhance the fairness and accuracy of annotations.

Acknowledgments

I want to thank Dr. Seals (my advisor) and Dr. Dozier (my co-advisor) for their support and valuable suggestions for this dissertation.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Our social lives have a significant impact on our lives in both positive and negative ways. It offers platforms for internet communication so that people may stay informed about a variety of news. People find it quite convenient to share some amusing, entertaining, and educational experiences in public. However, there are numerous instances of what are known as 'hate speeches' which are public remarks that promote violence or express hatred and injure people or organizations in different ways.

The necessity for the analysis and moderation of hate speech, at least in its most dangerous forms, has grown in recent years, and much effort has been put into the research and development of automatic methods to address it [124, 28, 108, 39]. Hate speech detection can be viewed from the standpoint of Natural Language Processing (NLP) as a text classification task: it is feasible to determine whether or not a text document created by a user (such as a post on a social network) contains hateful content [108]. In addition, other qualities could be investigated and analyzed, such as whether the text calls for violent action or not, if it is directed at a specific person or group, or which traits are the root of the attack [7].

Different type of hate speeches like irony, sexism, discriminatory ideas and sentiments, particularly those directed at women, are frequently found online and on social media. This typically indicates the presence of additional hazardous content types like sexist comments. Sexism is one of the most critical difficulties because it may greatly harm users. As an illustration, objectification, a kind of sexism, has been linked to serious risks for eating disorders, unipolar depression, and sexual dysfunction [42]. A study published by Fox et al. [41] requested participants to retweet or post tweets with sexist content before conducting a set of

activities meant to uncover sexist behaviors. Additionally, since the anonymity of Twitter can encourage people to demonstrate an even greater sexist behavior, the authors came to the conclusion that the users who were protected by the anonymity of a social media profile were more likely than non-anonymous users to exhibit aggressive sexism.

Sexism is defined by the Oxford English Dictionary as *prejudice, stereotyping or discrimination, typically against women, on the basis of sex*.[1] Sexist attitudes and languages undervalue the contribution of women. Furthermore, considering that a sizeable portion of internet users, particularly those who utilize social networks, are teenagers, the rise in online sexism necessitates urgent research and social debate that results in action [103].

However, detecting online hate speech may be difficult, as it may be expressed in very different forms. In this study, we focus mainly on using Natural Language Processing (NLP) methods and state-of-the-art models for hate speech detection. We seek to solve the issue of hate speech identification and classification in social media posts using a fully autonomous, NLP technique. This is a pressing issue that has not been adequately addressed previously [40, 30, 104], and the suggested solution is scalable and does not rely on inputs from humans directly.

The aim of this study also includes exploring how external features can be used to identify and categorize online hate speech. Numerous attributes, as well as the personalities and actions of individuals who express hate speech, have been studied in relation to their consequences. However, there aren't many studies that address the multi class classification problem in this context. In the specific context of our work, we aim to investigate how injection of user gender information with a fully autonomous NLP technique can improve identifying and categorizing sexism in social media posts. This is an urgent problem that has not been fully addressed in the past [40, 30, 104]. The suggested method is scalable and does not directly rely on input from humans.

This study also explores utilisation of Large Language Models in hate speech detection. In this context, we introduce a novel dataset OffensiveLang, a community based implicit offensive language dataset generated by ChatGPT containing data for 38 different target groups. Despite

[1] `https://www.oed.com/`

limitations in generating offensive texts using ChatGPT due to ethical constraints, we present a prompt-based approach that effectively generates implicit offensive languages. To ensure data quality, we evaluate our data with human. Additionally, we employ a prompt-based Zero-Shot method with ChatGPT and compare the detection results between human annotation and ChatGPT annotation. We utilize existing state-of-the-art models to see how effective they are in detecting such languages.

Lastly, we investigated annotator biases in large language models (LLMs) for hate speech detection and proposed mitigation strategies. Our study focused on four key categories of bias: gender, racial, religious, and disability. Specifically, we selected target groups that are highly vulnerable within these categories to explore the extent of annotator biases. We also conducted a detailed analysis of the possible reasons behind these biases by examining the data being annotated. Our work serves as a critical guide, aiming to steer researchers towards harnessing the potential of LLMs for data annotation, thereby facilitating future advancements in this essential domain.

Chapter 2

Related Work

Given the accessibility of the internet and the widespread use of social media, hate speech has become a significant issue online. Detecting various forms of hate speech, including irony, stereotypes, and sexism, has proven to be a challenging task that has evolved significantly over the years. While recent studies have focused on identifying hate speech, such as racist or xenophobic content, there has been less attention on detecting sexism, particularly hate speech directed towards women [103]. However, theories and techniques from general hate speech research can be applied to this problem [103]. In this section, we review past studies on hate speech detection, including related research in this area.

Automatically detecting hate speech is a complex task. Basic internet data filtering methods, such as simple 'bad word' lists, are not as effective for hate speech as they are for profanity. Hate speech is highly context-dependent; the same phrase can be interpreted differently depending on various factors, including the individuals involved, the timing of the comment, and the surrounding media [108].

Barbieri and Saggion [6] approached the automatic detection of irony as a classification problem, developing a model that used linguistic variables such as frequency, contrasts between written and spoken language, attitudes, ambiguity, intensity, synonymy, and structure to detect irony on Twitter. Nayel et al. [78] built a linguistically motivated feature set from tweets tagged with irony-related hashtags, achieving better performance than the bag-of-words approach.

Reyes and Rosso [102] focused on identifying components crucial for recognizing irony in English customer reviews, using features like n-grams, POS n-grams, humorous profiling,

positive/negative profiling, affective profiling, and pleasantness profiling. Their model showed competitive performance using three different classifiers [78].

Teh et al. [117] investigated the use of coarse language for detecting hate speech, categorizing 500 YouTube comments into eight hate speech categories based on profanity. Similar studies have focused on hate speech detection [117, 16], social media abuse [13, 79], fake news on Twitter [97], and cyberbullying [16]. Several publications and shared tasks have also focused on author profiling based on tweets [97, 98, 84].

Hernandez et al. [58] developed a model for irony detection on Twitter by framing it as a classification problem, evaluating it on various Twitter corpora with samples of ironic and non-ironic messages, showing good performance across different datasets [78].

Durlich et al. [37] presented the KLUEnicorn system, which used a Naive Bayes classifier to create word embeddings from adverb categories, named entities, and semantic and lexical data. A comprehensive review compared various supervised classification techniques, including Randomizable Filtered Classifier (RFC), Bayesian Network (BayesNet), and IBk [5].

Mohammad et al. [76] investigated the importance of sentiment expressed in a text for stereotype stance detection. They annotated the overall sentiment expressed in each instance in the SemEval-2016 Task 6 dataset, using features such as n-grams, char-grams, sentiment features from various lexica, and additional tweet-specific features. By combining these features with a support vector machine classifier, they achieved superior performance.

Rajadesingan and Liu [96] used a semi-supervised framework alongside a supervised classifier to detect users' perspectives on stereotype stance in tweets, employing a retweet-based label propagation method. This method categorized tweets as 'for' or 'against' based on their alignment with the values of surrounding labels.

Raghavan et al. [95] used a label propagation technique for community discovery, where each node adopted the label of the majority of its immediate neighbors, demonstrating exceptional performance in unsupervised settings.

Recent methods for fine-tuning language models to obtain meaningful phrase embeddings have shown promise [12, 101]. Rabelo et al. [90] used the universal sentence encoder, TFIDF, and a support vector machine to outperform many models in the case law retrieval challenge,

suggesting that combining TFIDF with BERT representation could be effective for identifying irony authors.

Early attempts to identify hate speech relied on bag-of-words (BOW) approaches [123, 124]. In 2012, abusive language was detected using machine learning-based classifiers [127] rather than pattern-based methods [47]. Various machine learning methods, including decision trees, logistic regression, and support vector machines, have been employed for hate speech detection [123].

In recent years, there has been significant interest in applying neural models to identify hate speech. In various natural language processing applications, these models often utilize deep learning techniques such as convolutional neural networks (CNNs) and long short-term memory networks (LSTMs) [87, 4, 136, 83].

Misogyny, while often related to sexism, specifically connotes expressions of rage and hatred toward women. While 'sexism' encompasses more covert forms of abuse and prejudice that can also significantly impact women, misogyny is a direct manifestation of such hostility. Glick and Fiske [49] distinguished between hostile sexism and benevolent sexism, with the former marked by overt hostility and the latter by seemingly positive but ultimately damaging attitudes. Sexism can manifest in various ways, including direct, indirect, descriptive, or documented behaviors (such as stereotyping, ideological differences, sexual aggressiveness, etc.). Therefore, misogyny is one form of sexism [72]. Much of the previous research has focused on detecting overt and explicit sexism, often neglecting more subtle forms [124, 43, 3, 82]. Addressing both overt and implicit sexism is crucial due to their prevalence and societal impact [57].

An underexplored area in hate speech detection is the use of user information. Understanding the personalities and behavioral tendencies of individuals who post harmful content online is pertinent for identifying hate speech. Chen et al. [17] proposed a Lexical Syntactic Feature architecture, arguing that the focus should be on the source of the content rather than treating messages as isolated instances. Waseem and Hovy [124] found that gender, among other extra-linguistic factors, improved hate speech recognition. Elise et al. [118] examined

the potential impact of various user features on hate speech classification. However, the scarcity or absence of user information poses challenges for many researchers in this field.

Wulczyn et al.'s [126] qualitative analysis of personal attacks in Wikipedia comments found that anonymity increases the likelihood of attacks, even though anonymous comments constituted less than half of all attacks. The study also noted that personal attacks tend to cluster over time, suggesting that one attack often incites another. Cheng et al. [19] examined antisocial behavior in online discussion forums by comparing banned users with non-banned users. They found that banned users focused on fewer threads, used less polite and more offensive language, and received more reactions than others.

Chatzakou et al. [15] explored user attributes to enhance the detection and classification of bullying and aggressive behavior on Twitter. They found that network-based characteristics, such as the number of friends and followers, reciprocity, and network rank, were most effective in identifying hostile user behavior. Langham et al. [65] developed a reliable and accurate classification of angry social media posts for hate speech identification, emphasizing the role of anger in promoting hate crimes. Modi et al. [75] evaluated six different hate speech detection techniques across social media platforms, highlighting the differences between methods from natural language processing, data mining, and machine learning.

Large language models have recently been used frequently for data generation. Yu et al. [128] introduced the ChatGPT-writtEn AbsTract dataset (CHEAT) to enhance the classification of human-written and ChatGPT-written abstracts. Alamleh et al. [2] differentiated between human-written and ChatGPT-written texts by collecting responses from computer science students and classifying the texts using various machine learning models. Chen et al. [18] released the OpenGPTText dataset containing rephrased content generated by ChatGPT and used RoBERTa and T5 models for classification, successfully distinguishing between human-written and ChatGPT-generated texts. Hartvigsen et al. [54] generated a large-scale machine-generated dataset for implicit hate speech detection, using GPT-3 to extract toxic and benign statements targeting 13 minority groups.

Most feature-based offensive language detection methods involve supervised text classification, typically requiring feature engineering. Razavi et al. [100] constructed a lexicon of

insulting and abusive language, assigning weights to each word to improve detection accuracy. Chen et al. [17] introduced the Lexical Syntactic Feature (LSF) method to detect derogatory content and offensive users on social media, incorporating lexical, stylistic, structural, and context-specific features alongside traditional machine learning models.

Oriola and Kotze [80] explored various machine learning techniques for detecting hate speech and offensive language in South African tweets. Their research highlighted the importance of English slur words in identifying offensive language. Similarly, Vargas et al. [119] developed an approach for detecting offensive language and hate speech on social media using a contextually annotated offensive lexicon, which proved effective across different languages.

Recent advancements have seen deep learning techniques and large-scale pretrained language models applied to offensive language detection. For instance, the BERT model [32] has been widely used. Liu et al. [70] fine-tuned BERT for transfer learning to detect offensive content in the SemEval-2019 Task 6.

Combining user information with textual data to enhance detection accuracy is another research area. Qian et al. [89] proposed a method that uses intra-user and inter-user representations, collecting and analyzing users' historical tweets and employing local sensitive hashing (LSH) for inter-user tweets. This combination significantly improved the F1-score.

The use of Large Language Models (LLMs) for identifying hate speech has gained traction. Chiu et al. [20] used few-shot learning with GPT-3 to detect sexist and racist texts. Li et al. [66] found that ChatGPT's performance in categorizing harmful content was comparable to expert annotations from Amazon Mechanical Turk. He et al. [56] enhanced toxic content detection accuracy using LLMs like GPT-3 and T5 alongside prompt learning.

Zhu et al. [134] utilized ChatGPT to re-annotate various datasets, including those for hate speech detection, revealing significant disparities with human annotations. Li et al. [66] also employed ChatGPT to categorize harmful comments, noting its effectiveness in detecting non-harmful comments. Huang et al. [59] used ChatGPT to classify implicit hate speech by framing prompts as binary questions, differing from ElSherief et al. [38], who defined specific classes like implicit hate, explicit hate, and non-hate.

The popularity of large-scale Pre-trained Language Models (PLMs) such as GPT [93], BERT [62], and RoBERTa [71] has led to increased acceptance of prompt-based learning. Studies have focused on using PLMs as repositories of implicit and unstructured knowledge [116] [29] [110]. Recent works have utilized prompts to guide PLMs in various NLP tasks, including sentiment classification, achieving notable performance in few-shot settings [44] [106] [107]. Additionally, research is emerging on applying prompts to visual-language models for computer vision tasks [133] [91].

The advent of large language models (LLMs) has revolutionized NLP tasks by enabling sophisticated and context-aware language understanding systems. Models such as BERT [32], GPT [92], and their variants have shown remarkable performance across text classification, language generation, and question answering tasks. These models leverage pre-training on large corpora followed by fine-tuning on task-specific data to capture intricate linguistic patterns and semantic relationships.

Recent research has leveraged LLMs for data annotation, utilizing their ability to understand and generate human-like text. Gururangan et al. [51] proposed a framework for generating natural language explanations for machine learning models, aiding in the annotation of model predictions with interpretable justifications. Raffel et al. [94] introduced a method for efficiently annotating speech data using GPT-2, significantly reducing annotation time compared to traditional manual labeling.

The interest in using LLMs as versatile annotators for various natural language tasks has been growing [64] [134] [135]. Wang et al. [122] demonstrated that GPT-3 could significantly reduce labeling costs by up to 96% for classification and generation tasks. Ding et al. [35] assessed GPT-3's effectiveness in labeling and data augmentation across classification and token-level tasks. Evidence suggests that LLMs can outperform crowdsourced annotators in certain classification tasks [48] [55].

Research into social biases in NLP models is extensive, addressing allocational and representational harms [8] [22]. Various methodologies have been proposed to assess and mitigate these biases in NLU [9] [31] [36] [10] [132] [114] and NLG tasks [112] [34]. Sun and Peng

[114] utilized the Odds Ratio (OR) [115] to quantify gender biases, while Sheng et al. [112] assessed biases in NLG model outputs. Dhamala et al. [33] extended this analysis with real-world prompts from Wikipedia. Various strategies to mitigate biases in NLG models have been proposed [113] [50] [68] [11], though their applicability to closed API-based LLMs like ChatGPT remains uncertain.

Chapter 3

Irony and Stereotype Spreading Author Profiling on Twitter

3.1 Introduction

[1] Irony is a profoundly pragmatic and versatile linguistic phenomenon. Its foundations typically lie beyond explicit linguistic patterns, relying instead on reconstructing contextual dependencies and latent meanings such as shared or common knowledge [61]. This makes automatic detection of irony a challenging task in natural language processing. In this study, we address the issue of profiling individuals who disseminate irony and stereotypes on social media, with a particular focus on Twitter. We specifically target authors who use irony to spread stereotypes, particularly those directed at women, immigrants, or the LGBTQ+ community. Our primary objective is to classify authors based on their use of ironic content to identify those who potentially propagate stereotypes. This classification serves as an initial step toward mitigating such behavior. In this work, we explore the following: 1) The phenomena of irony and stereotypes in English tweets and potential methods for their detection, and 2) Two specific representations for addressing this issue: i) BERT and ii) Term Frequency Inverse Document Frequency (TFIDF) combined with BERT. The classification task was subsequently implemented using a Logistic Regression classifier.

3.2 Dataset

The task includes both irony detection and stereotype stance detection problem. The data was taken from PAN 22 author profiling task [81]. For the irony detection task, the dataset contained

[1] This work [26] is a part of the Profiling Irony and Stereotype Spreaders on Twitter (IROSTEREO) shared task for PAN at CLEF 2022.

tweets of 600 authors each having 200 tweets. It was split into two categories: 1) validation dataset containing tweets of 420 authors and 2) test dataset containing tweets of 180 authors. The validation dataset is a balanced dataset (50% of them were irony and 50% of them were non irony) containing total 84000 tweets (420 authors each having 200 tweets). This dataset is used for the training purposes. For testing, 180 authors each containing 200 tweets are used. The training set is balanced, i.e. out of 420 authors 210 are irony and 210 are not irony.

For the stereotype stance detection subtask, the dataset contained tweets of 200 authors each having 200 tweets. It was again split into two categories: 1) validation dataset containing tweets of 140 authors and 2) test dataset containing tweets of 60 authors. The validation dataset is an imbalanced dataset containing 28000 tweets (140 authors each having 200 tweets). This dataset is used for the training purposes. For testing, 60 authors each containing 200 tweets are used. The goal of this subtask is to detect the stance of how stereotypes are used by ironic authors, if in favour or against the target. Table 2 shows the details of the dataset. The training set is imbalanced, i.e. out of 140 authors 94 are AGAINST and 46 are INFAVOR.

3.3 Methodology

Here we explain our methods used for this study. We detail each of the feature spaces in the following lines:

3.3.1 BERT

The design of a neural encoder for natural language sequences has been changed by Transformer [120], a sequence transduction model based on attention mechanisms. The transformer architecture allows sequential data to be learned. To improve on largely unidirectional language model training, Devlin et al. [32] developed Bidirectional Encoder Representations from Transformers (BERT). BERT makes deep bidirectional language encoding training achievable by employing the masked language modeling (MLM) loss [21]. BERT employs next-sentence prediction (NSP), an extra loss for pre-training that aims to learn high-level linguistic coherence by predicting whether or not two text segments should come sequentially in the original text [21].

The sentence had to be tokenized first before the embeddings could be created. Point to be noted, BERT can only handle sentences with a length of 512 tokens or less. BERT's authors advise using the BERT Base Uncased model in the majority of cases unless it is clear that using a case-sensitive model will be beneficial to the task [74]. Using 1s and 0s to discriminate between the two sentences, BERT is trained on and anticipates sentence pairs [74]. That is to say, we must indicate whether each token in "tokenized text" fits in sentence 0 (a string of 0s) or sentence 1 (a series of 1s). We constructed a vector of 1s for each token in our input sentence since single-sentence inputs just need a string of 1s for our needs [74]. We then called the BERT model after converting our data to torch tensors. The number of layers (13 layers), the batch number (1 sentence), the word/token number (22 tokens in our sentence), and the hidden unit/feature number make up the complete set of hidden states for this model (768 features). The first element represents the input embeddings, and the remaining elements are the outputs of each of the 12 layers of BERT, hence the layer number is 13 [74].

When sending several sentences to the model at once, the batch size, the second dimension, is employed. There would be one batch total. We had 13 distinct vectors, each of which was 768 bytes long, for each token in our input. We combined the final four layers to produce a word vector with a length of 3072 (4×768 = 3072) per token. We calculated the average of the second-to-last hidden layer of each token to produce a single vector of 768 length for our complete text[74].

3.3.2 TFIDF

We apply the well-known Term Frequency Inverse Document Frequency (TFIDF) weighting system in our methodology to extract traditional features. TFIDF is a combination of two different terms: Term Frequency (TF) and Inverse Document Frequency (IDF) [88]. The term TF is used to calculate the frequency of a term in a document [52]. The term frequency for a term t and a document d is defined by (equation 3.1):

$$tf_{d,t} = \frac{n_{d,t}}{|d|} \tag{3.1}$$

where $n_{d,t}$ is the number of occurrences of the term t in the document d. The term frequency $tf_{d,t}$ is then the number of occurrence of the term t in document d divided by the total number of tokens in the document.

The inverse document frequency of a term t in the whole collection is (equation 3.2):

$$idf_t = log\frac{|D|}{|d : t \in d|} \tag{3.2}$$

where $|D|$ is the number of classes in the classification problem and $|d : t \in d|$ is the number of document(s) where the term t appears.

When calculating a document's term frequency, it can be seen that the algorithm evaluates all keywords similarly, regardless of whether they are stop words or not, which is incorrect because all keywords have varying relevance [52]. The inverse document frequency method gives less weight to often occurring words and more weight to infrequently occurring terms [52]. Mathematically, TFIDF is the multiplication of term frequency (TF) and inverse document frequency (IDF). The formula that is used to compute the TFIDF of term t present in document d is (equation 3.3):

$$tfidf_{d,t} = tf_{d,t} * idf_t = \frac{n_{d,t}}{|d|} * log\frac{|D|}{|d : t \in d|} \tag{3.3}$$

TFIDF's purpose is to lessen the impact of less informative tokens that appear frequently in a data corpus [45]. We used TfidfVectorizer features from scikit-learn to perform the TFIDF task [85]. Table 3.1 shows the TFIDF parameter values used for the tasks.

Task	***TFIDF_max_df***	***TFIDF_min_df***
Irony spreading author profiling	0.70	1
Stereotype spreading author profiling	0.95	1

Table 3.1: TFIDF parameters

3.3.3 BERT combined with TFIDF

TFIDF is used to evaluate how relevant a word is to a document in a collection of documents. The TFIDF score can be fed to the Bert model to improve the prediction performance. In

order to produce a deeper and more insightful quantitative representation of the data, we used this embedding approach. We combined TFIDF with word embedding. We put a threshold of less than 1000 words while implementing the TFIDF. The idea is to preserve the grammatical regularities in each document intact.

3.4 Results and Discussion

To understand the efficiency of our models, first we split the validation data into training and testing set. Out of 420 authors, we used 336 of them as training and remaining 84 were used for testing. The datasets were split balanced, i.e. for both the training and testing data, 50% of them were irony and 50% of them were non irony. We combined all 200 tweets of each person and treated it as a single string. It basically created 420 strings for 420 authors each string containing the combination of 200 tweets of each author. Initially we were only concerned about the accuracy of different machine learning models for only the BERT representation. The strings were then converted to numeric values using the BERT. We implemented the following five machine learning algorithms to check the best accuracy: KNN, SVM, Decision Tree, Naive Bayes and Logistic Regression and it was seen that the Logistic Regression was proved to be the best in terms of efficiency and accuracy. To measure the accuracy of an algorithm, we used the formula in equation 3.4.

$$Accuracy = \frac{TrueLabel}{TrueLabel + FalseLabel} \tag{3.4}$$

Where True Label refers to correct prediction and False Label refers to incorrect classification. The classification results obtained from the five algorithms on the validation dataset are given in Table 3.2.

Algorithm	***Accuracy(%)***
KNN	89.2
SVM	88
Decision Tree	77.3
Naive Bayes	91.6
Logistic Regression	92.8

Table 3.2: ML algorithms implemented on ironic author profiling validation dataset

The BERT implementation method was then implemented on the testing dataset of tweets of 180 authors after training it with the validation dataset tweets of 420 authors. It was seen that the classification results were not very promising. However, after combining the BERT representation with TFIDF, the classification results were improved from 38% to 67% which was an increase of around 76%. The classification results on the test dataset are shown in Table 3.3.

Representation	***Classifier***	***Accuracy(%)***
BERT	Logistic Regression	38
BERT-TFIDF	Logistic Regression	67

Table 3.3: Accuracy on ironic author profiling test dataset

The similar method was implemented to address the subtask of stereotype stance detection used by ironic authors either in favour or in against. Unlike the dataset used for ironic author classification, the dataset used for stereotype stance detection was smaller in size and also imbalanced. The classification result of stereotype stance detection problem using our model is shown in Table 3.4. We obtained an overall macro F1 score of 0.45 and F1 score of 0.19 InFavour.

Representation	***Classifier***	***Accuracy(%)***
BERT-TFIDF	Logistic Regression	58

Table 3.4: Accuracy on stereotype favouring author profiling test dataset

The BERT representation alone did not yield high classification accuracy. TF-IDF enhances the impact of informative tokens that frequently appear in the data corpus. By integrating BERT embeddings with TF-IDF features, the model leverages both BERT's contextual understanding and TF-IDF's term importance evaluation. TF-IDF emphasizes key terms indicative of irony, while BERT captures deep contextual information. This combination substantially improves the model's performance in detecting irony in tweets. These findings suggest that despite BERT's strength, augmenting it with additional features like TF-IDF is crucial for specific tasks such as irony detection. The results highlight the significance of feature engineering and the necessity to tailor models to the specific characteristics of the data and task.

3.5 Summary

In this study, we presented a method for profiling irony and stereotype spreaders on Twitter to address the challenge of detecting authors who spread irony. Additionally, we tackled a subtask focused on stereotype stance detection, which involves determining whether ironic authors use stereotypes favorably or unfavorably towards the target. Our approach involved implementing a BERT model, and subsequently combining BERT with TFIDF for tweet representation. We then used a logistic regression classifier to differentiate between irony-spreading and non-irony-spreading authors. BERT was employed to convert text data into numerical format, while TFIDF highlighted the most significant tokens. Our findings indicate that combining BERT representation with TFIDF markedly improved classification results. In conclusion, we have demonstrated effective techniques for classifying irony and stereotype spreaders on Twitter.

Chapter 4

Explainable Detection of Online Sexism

4.1 Introduction

[1]In this chapter, we further investigate the detection of online sexism. In the specific context of this work, we seek to solve the issue of sexism identification and classification in social media using a fully autonomous, NLP technique. This is a pressing issue that has not been adequately addressed previously [40, 30, 104], and the suggested solution is scalable and does not rely on inputs from humans directly. Our contributions: 1) Propose a unified model (fine-tuned RoBERTa), to address all three tasks mentioned in figure 4.1. 2) Evaluate the efficacy of our model by conducting a comparative analysis with other existing models.

4.2 Task Description

The task [63] deals with identifying online sexism in English. The task contains three subtasks: Task A, Task B and Task C, explained in Figure 4.1.

4.3 Dataset

The data was taken from SemEval 2023 Expalinable Detection of Online Sexism task [63]. The dataset with labels had 20,000 entries, 10,000 of them were drawn from Reddit and 10,000 from Gab [63]. For labeling them, three trained annotators first labeled each entry, and one of two experts then decided on any differences. The task contains three tasks: task A, task B and task C. When all of the annotators agreed on one label for Task A, that label is considered

[1]This work [25] is a part of the SemEval 2023 Task 10: Explainable Detection of Online Sexism.

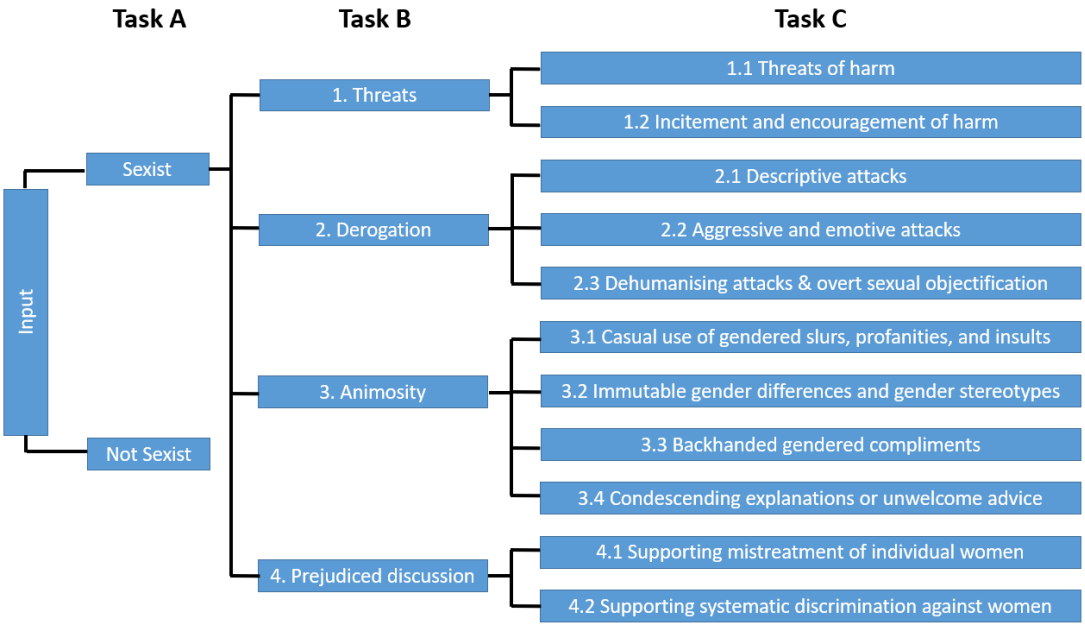

Figure 4.1: Task Description

to be the gold label. One of the experts evaluated the entry and chose the gold label if there was any disagreement. When two or more annotators agreed on a label for Tasks B and C, the label was considered the gold label; however, if there was a three-way tie, one of the experts determined the gold label. The experts and annotators were all self-identified women. The detailed definitions of the categories of the 3 tasks are given below:

Task A: The initial tier of the taxonomy categorizes content into two distinct groups: sexist and non-sexist. Sexist content encompasses any form of abuse, whether implicit or explicit, directed at women due to their gender. This also includes instances where gender is combined with other identity attributes.

Task B: In the second level of taxonomy, sexist content is disaggregated into four distinct categories, both conceptually and analytically. These categories are as follows: 1. Threats, Plans to Harm, and Incitement: This category includes language where an individual expresses intent to harm or encourages others to take harmful actions against women. This encompasses threats of physical, sexual, or privacy-related harm and incitement of serious violence against

women. 2. Derogation: This category encompasses language that explicitly degrades, dehumanizes, demeans, or insults women. It includes negative descriptions and stereotypes, objectification of women's bodies, strong negative emotive statements, and dehumanizing comparisons. This category covers derogatory remarks directed at specific women as well as women in general. 3. Animosity: This category includes language that expresses implicit or subtle sexism, stereotypes, or descriptive statements. It also includes benevolent sexism, which is sexism framed as a compliment. 4. Prejudiced Discussion: This category involves language that denies the existence of discrimination and justifies sexist treatment. It includes the denial and justification of gender inequality, excusing mistreatment of women, and promoting the ideology of male victimhood.

Task C: The third level of taxonomy breaks down each category of sexism into eleven detailed sexism vectors as follows: 1.1 Threats of Harm: Expressing an intent, willingness, or desire to harm an individual woman or group of women. This encompasses, but is not limited to, physical, sexual, emotional, or privacy-related harm. 1.2 Incitement and Encouragement of Harm: Encouraging or inciting an individual, group, or general audience to harm a woman or group of women. This includes language that seeks to rationalize and/or justify harming women to others. 2.1 Descriptive Attacks: Characterizing or describing women in a derogatory manner. This includes negative generalizations about women's abilities, appearance, sexual behavior, intellect, character, or morals. 2.2 Aggressive and Emotive Attacks: Expressing strong negative sentiments against women, such as disgust or hatred. This can involve direct descriptions of the speaker's subjective emotions, baseless accusations, or the use of gendered slurs, profanities, and insults. 2.3 Dehumanizing Attacks and Overt Sexual Objectification: Derogating women by comparing them to non-human entities such as vermin, disease, or refuse, or overtly reducing them to sexual objects. 3.1 Causal Use of Gendered Slurs, Profanities, and Insults: Using gendered slurs, profanities, and insults casually, without the intention of directly attacking women. Only terms traditionally describing women are considered. 3.2 Immutable Gender Differences and Gender Stereotypes: Asserting natural, essential, or immutable differences between men and women. This can also include using women's traits to attack men. 3.3 Backhanded Gendered Compliments: Giving seemingly positive comments to women that

actually belittle them or imply their inferiority. This includes reducing women's value to their attractiveness or sexual desirability, or implying that women are inherently frail, helpless, or weak. 3.4 Condescending Explanations or Unwelcome Advice: Offering unsolicited or patronizing advice to women on topics they are knowledgeable about. 4.1 Supporting Mistreatment of Individual Women: Expressing support for the mistreatment of individual women by denying, understating, or justifying such behavior. 4.2 Supporting Systemic Discrimination Against Women: Expressing support for systemic discrimination against women as a group by denying, understating, or justifying such discrimination.

For task A, 14000 posts for training, 2000 for validation and 4000 for testing were used. The training dataset was imbalanced. Out of 14000 posts used for training, 3398 posts were labeled as 'sexist', remaining 10602 as 'non sexist'. For tasks B and C, 486 posts for validation and 970 posts for testing were used.

4.4 Methodology

Recently, Natural Language Processing has seen a rise in popularity of Pretrained Language Models (LMs). A pre-trained language model has the drawback of taking a long time during sentence pair regression processes like clustering and sentence similarity analysis [111]. The sentence can be embedded to remedy the issue. Recently, many sentence embedding techniques with varied generating mechanisms have been proposed. Our model used in this study is described below.

4.4.1 RoBERTa

BERT was significantly undertrained for few specific tasks. In order to address this weakness, a new recipe for training BERT models termed RoBERTa was proposed [71]. RoBERTa stands for Robustly optimized BERT approach. RoBERTa can match or outperform the performance of all post-BERT approaches. The changes in RoBERTa consist of the following: (1) training the model for a longer period of time with larger batches of data; (2) eliminating the objective of next sentence prediction; (3) training on longer sequences; and (4) dynamically altering

the masking pattern used on the training data [71]. In particular, dynamic masking, FULL-SENTENCES without NSP loss, big mini-batches, and a bigger byte-level BPE are used to train RoBERTa [71].

RoBERTa is a multilingual model with Transformer as the primary structure, that was used for the representation. The model has 12 layers with 768 output dimensions. After obtaining the embedding vectors of the texts, RoBERTa was improved to make it better suited for the subsequent task of identifying hate speech. The 12 layers that makeup RoBERTa, each learn a distinct type of semantic data. In general, more word level semantic information is learned when the layers are thinner. More broad semantic knowledge is learned as one delves further into the levels. For binary classification tasks like subtask A, global semantic information is more beneficial. The RoBERTa model's hidden layer has 12 layers of Transformer and 768 dimensions. The 12th layer was removed as the dimension was (0,2) and was spliced it with a vector (Tc) where Tc's shape was [32,768] and the hidden vector of the 12th layer was [32,60,768]. The resulting data was then passed to the classifier and the results were sent to Softmax. By incorporating the transformed data into the model, both tasks were trained. Similar activities can provide useful information for multitask learning. We used RoBERTa large for our research. 24 transformer layers, each with 16 self-attention heads and 1,024 hidden units, make up the RoBERTa large model.

4.4.2 Fine-Tuned RoBERTa

We utilized a multi-task learning approach to address all three tasks simultaneously with a single model. This was achieved by encoding the initial representation of a post using Roberta and setting three distinct Multilayer Perceptrons (MLPs) corresponding to the three sub-tasks. Our model can be viewed as a hard multi-task learning paradigm where all three heads share the 24 layers defined in Roberta large. The average loss obtained from each head is used to backpropagate and adjust the parameters of the model, resulting in a reduction of the final model size by one-third compared to the separated version without sacrificing performance.

4.5 Results and Discussion

4.5.1 Task A

We first implemented our model on the validation data and then compared the performance of it with three other models: sentence-BERT, BERT and BERT representation combined with TFIDF. It was seen that our model the fine-tuned RoBERTa had the highest F1 score. Table 4.1 shows the F1 scores of the models we used on validation data. Out of all the models listed here, fine tuned RoBERTa had significantly higher F1 score than the other models on the development data.

Representation	***Macro-F1***
SBERT	67.26
BERT	68.94
BERT-TFIDF	71.36
Fine tuned RoBERTa	83.64

Table 4.1: F1 scores of different models on the development dataset for Task A

Figure 4.2 shows the confusion matrix of our prediction with true label on task A development data. The number of correct predictions of 'not sexist' and 'sexist' comments are 1423 and 345 respectively which makes total 1768 correct predictions out of 2000 data.

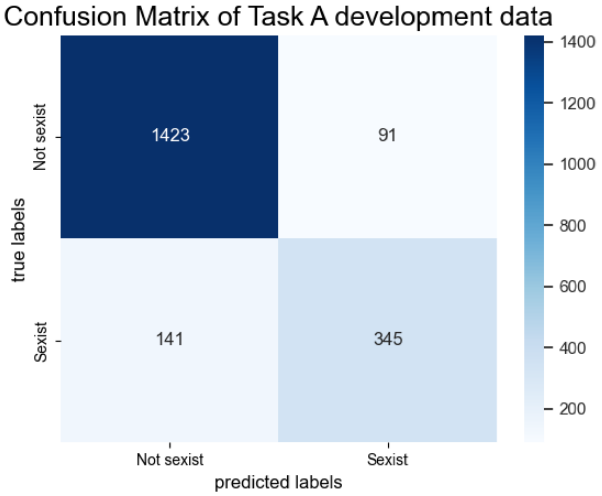

Figure 4.2: Confusion Matrix of Task A development data

The test dataset contained 4000 posts. The F1 score of our model on the test dataset for task A is shown in Table 4.2.

Representation	*Macro-F1*
Fine tuned RoBERTa	79.43

Table 4.2: F1 scores of our model on the test dataset for Task A

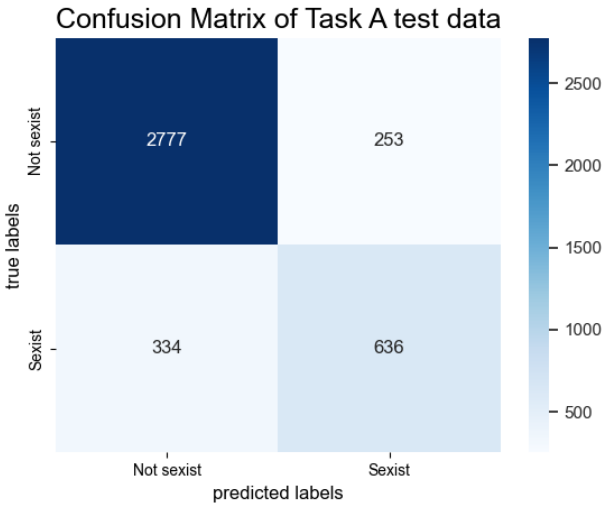

Figure 4.3: Confusion Matrix of Task A test data

Figure 4.3 shows the confusion matrix of our prediction with true label on task A test data. The number of correct predictions of 'not sexist' and 'sexist' comments are 2777 and 636 respectively which makes total 3413 correct predictions out of 4000 data.

4.5.2 Task B

For task B also we first implemented our model on the validation data and then compared the performance of it with the previously mentioned other three models. Table 4.3 shows the F1 score of the models we used on validation data.

Representation	*Macro-F1*
SBERT	53.74
BERT	54.58
BERT-TFIDF	50.56
Fine tuned RoBERTa	65.88

Table 4.3: F1 scores of different models on the development dataset for Task B

Figure 4.4 shows the confusion matrix of task B on the development dataset where number 1 refers to category 'Threats', number 2 refers to category 'Derogation', number 3 refers to

category ‘Animosity’ and number 4 refers to category ‘Prejudiced Discussion’. The number of correct predictions of the four categories are 31, 182, 84 and 32 respectively.

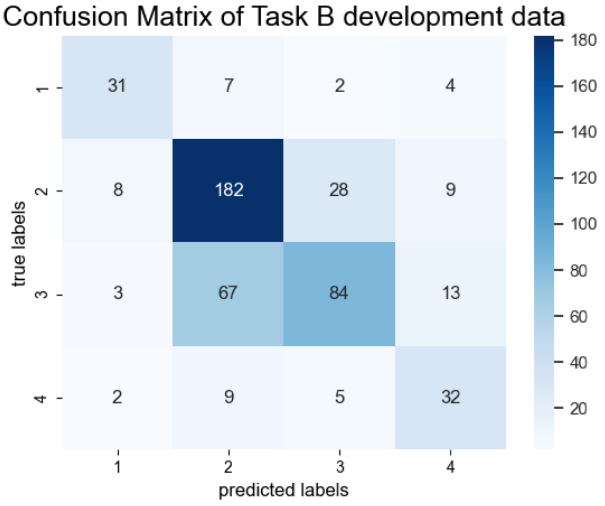

Figure 4.4: Confusion Matrix of Task B development data

The test dataset contained 970 posts. The F1 score of our model on the test dataset is shown in Table 4.4.

Representation	***Macro-F1***
Fine tuned RoBERTa	61.91

Table 4.4: F1 scores of our model on the test dataset for Task B

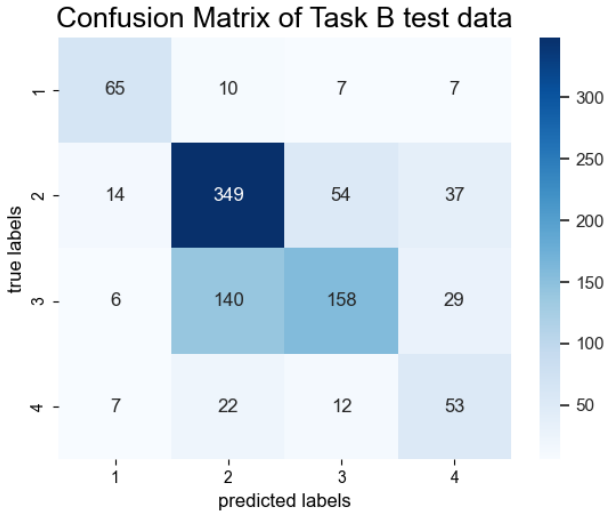

Figure 4.5: Confusion Matrix of Task B test data

Figure 4.5 shows the confusion matrix of task B on the test dataset. The number of correct predictions of the four categories are 65, 349, 158 and 53 respectively.

4.5.3 Task C

Similar to task B, for task C also we first implemented our model on the validation data and then compared the performance of it with the previously mentioned other three models. It is interesting to see that not only binary classification, but our model was efficient for multi class classification task also. Table 4.5 shows the F1 score of the models we used on validation data.

Representation	***Macro-F1***
SBERT	32.74
BERT	32.14
BERT-TFIDF	26.64
Fine tuned RoBERTa	33.20

Table 4.5: F1 scores of different models on the validation dataset for Task C

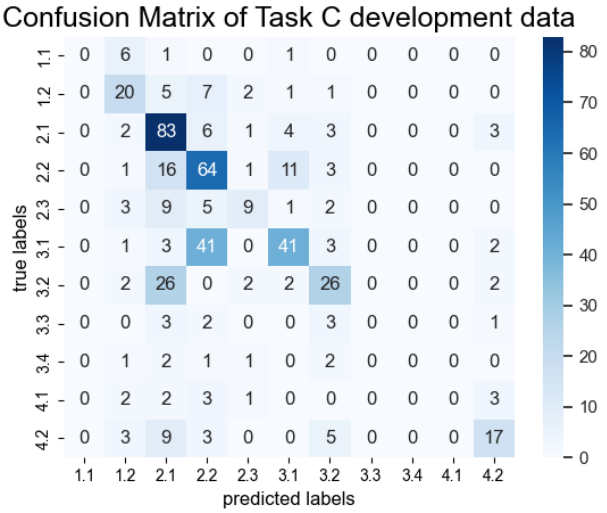

Figure 4.6: Confusion Matrix of Task C development data

Figure 4.6 shows the confusion matrix of task C on the development dataset where number 1.1 refers to category 'Threats of harm', number 1.2 refers to category 'Incitement and encouragement of harm', number 2.1 refers to category 'Descriptive attacks', number 2.2 refers to category 'Aggressive and emotive attacks', number 2.3 refers to category 'Dehumanisation and overt sexual objectification', number 3.1 refers to category 'Casual use of gender slurs, profanities and insults', number 3.2 refers to category 'Immutable gender stereotypes', number 3.3 refers to category 'Backhanded gendered compliments', number 3.4 refers to category

'Condescending explanations or unwelcome advice', number 4.1 refers to category 'Supporting mistreatment of individual women' and number 4.2 refers to category 'Supporting systemic discrimination against women'. The number of correct predictions of the eleven categories are 0, 20, 83, 64, 9, 41, 26, 0, 0, 0, and 17 respectively.

The test dataset contained 970 posts. The F1 score of our model on the test dataset is shown in Table 4.6.

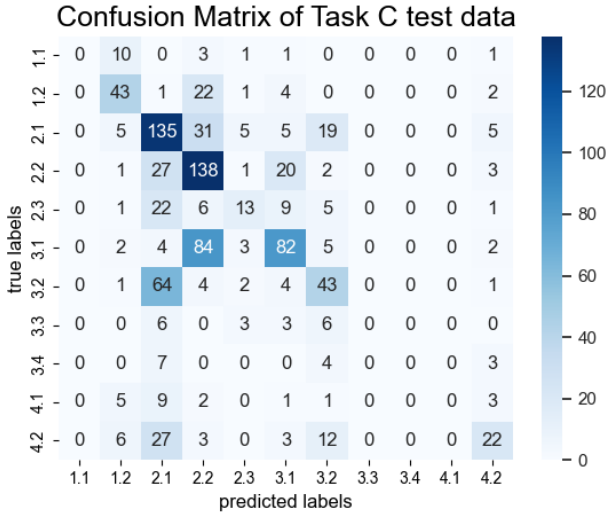

Figure 4.7: Confusion Matrix of Task C test data

Representation	***Macro-F1***
Fine tuned RoBERTa	29.9

Table 4.6: Accuracy on Development data

Figure 4.7 shows the confusion matrix of task C on the development dataset. The number of correct predictions of the eleven categories are 0, 43, 135, 138, 13, 82, 43, 0, 0, 0, and 22 respectively.

4.6 Summary

In this study, we introduced a model designed to tackle the problem of sexism detection on Gab and Reddit data. The task was divided into three subtasks: binary sexism classification, categorization of sexism types, and fine-grained vector classification of sexism. To address these challenges, we implemented a fine-tuned RoBERTa model and compared its performance with

several other models, including BERT, SBERT, and a combination of BERT with TFIDF. Our results demonstrated that our model achieved the highest Macro-F1 score among the models implemented. In conclusion, we have demonstrated an effective technique for online sexism detection.

Chapter 5

Online Sexism Detection and Classification by Injecting User Gender Information

5.1 Introduction

[1]To further investigate online sexism detection, we aim to explore the role of gender information of social media users in identifying and categorizing sexist content. While numerous studies have examined the attributes, personalities, and behaviors of individuals who engage in hate speech, there is a notable gap in addressing the multi-class classification problem within this context. Specifically, our research seeks to determine how incorporating user gender information with a fully autonomous NLP technique can enhance the detection and categorization of sexism in social media posts. This urgent issue has not been adequately addressed in previous studies [40, 30, 104]. Our objective is to understand the impact of user gender information on the three sexism detection tasks outlined in the previous chapter.

5.2 Methodology

This section describes some of the natural language processing and machine learning models used in this work.

5.2.1 SBERT

Sentence-BERT (SBERT) was employed for embedding. SBERT is a BERT [32] variation that uses siamese and triplet network architectures to produce semantically important sentence

[1]This work [24] was presented at the 2023 IEEE AIBThings conference.

embeddings [101]. Here, the text documents are broken up into paragraphs before being represented as text documents using the average of SBERT paragraph embeddings. The text documents are then grouped according to the SBERT representations of each document with the maximum cosine similarity [109].

Using a variation of the masked language modeling aim used in the pretraining of the BERT model, the SBERT model is pretrained on a vast volume of text data. The model can be fine-tuned on particular downstream goals like text classification, question answering, or semantic similarity tasks after pretraining. On various benchmark datasets for tasks like phrase categorization and semantic similarity, SBERT has demonstrated state-of-the-art performance, demonstrating its excellent efficacy in capturing sentence semantics. Due to its capacity to produce high-quality sentence embeddings, it is a preferred option for many NLP applications, including chatbots, recommendation systems, and hate speech detection.

5.2.2 Word2vec

Popular sequence embedding technique Word2vec converts plain text into distributed vector representations of emotions [69]. It has been widely utilized as a foundation for predictive models in semantic and information retrieval tasks because it can capture contextual word-to-word interactions in a multidimensional space. Continuous Bag of Words (CBOW) and skip-gram [131] are two distinct parts of the Word2vec method. While the skip-gram component infers the context words when given an input word like [86], the CBOW component infers the target word when given the context words.

5.2.3 Injection of Gender Information

We used the SemEval 2023 online sexism detection dataset [63] for sexism detection task and the PAN 2015 gender prediction English dataset [99] for the pretraining in this work. We used SBERT for representation of the both PAN and SemEval data. The embedding length was 384 for each post. We used Word2Vec of length 20 to represent the gender either as ‘male’ or as ‘female. We combined the gender embed with the text representation which made the total embed length 404 for each post. We then used the logistic regression classifier for

the classification. Figure 5.1 describes our proposed architecture. We did it for both with and without including the gender embed. The results of the experiments are discussed in the following section.

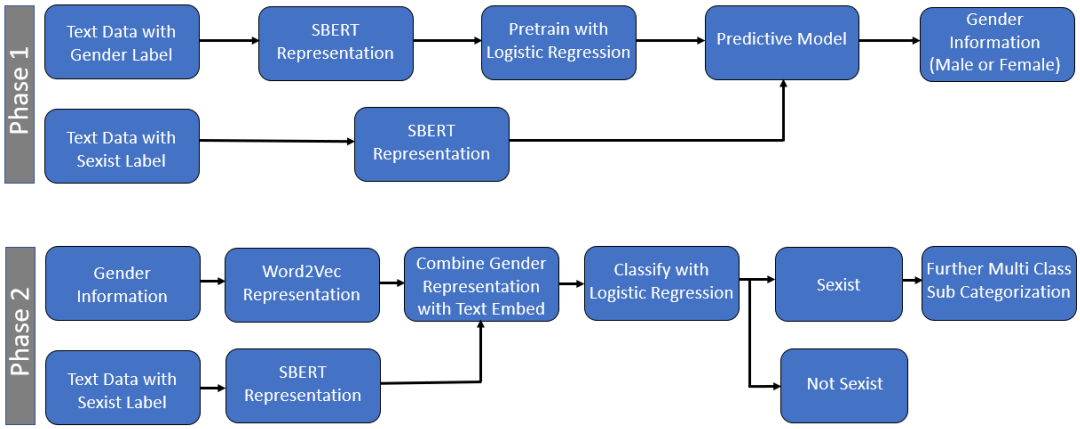

Figure 5.1: Overview of the Proposed Architecture

5.3 Results and Discussion

This section presents the results of our experiment and discusses the effects of incorporating user gender information with text data. To measure the accuracy, we used the following formula (equation 5.1):

$$Accuracy = \frac{TrueLabel}{TrueLabel + FalseLabel} \tag{5.1}$$

Where TrueLabel refers to correct prediction and FalseLabel refers to incorrect prediction. Table 5.1 shows the accuracy of the 3 tasks on the development dataset. It can be seen that for all the 3 tasks, inclusion of the gender embed increased the accuracy.

Embedding type	***Task***	***Accuarcy(%)***
Embedding without gender	Task A	78.70
Embedding with gender	Task A	80.14
Embedding without gender	Task B	62.96
Embedding with gender	Task B	63.47
Embedding without gender	Task C	58.84
Embedding with gender	Task C	60.28

Table 5.1: Accuracy on Development data

Embedding type	*Task*	*Accuarcy(%)*
Embedding without gender	Task A	79.38
Embedding with gender	Task A	80.97
Embedding without gender	Task B	63.60
Embedding with gender	Task B	64.43
Embedding without gender	Task C	60.00
Embedding with gender	Task C	60.61

Table 5.2: Accuracy on Test data

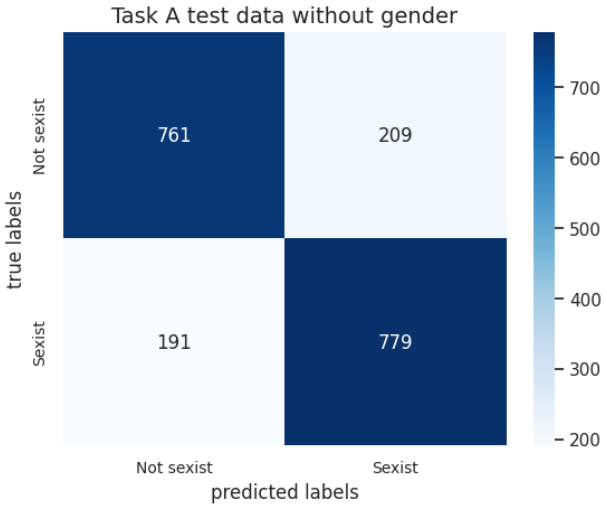

Figure 5.2: Confusion Matrix of Task A test data without inclusion of gender embed

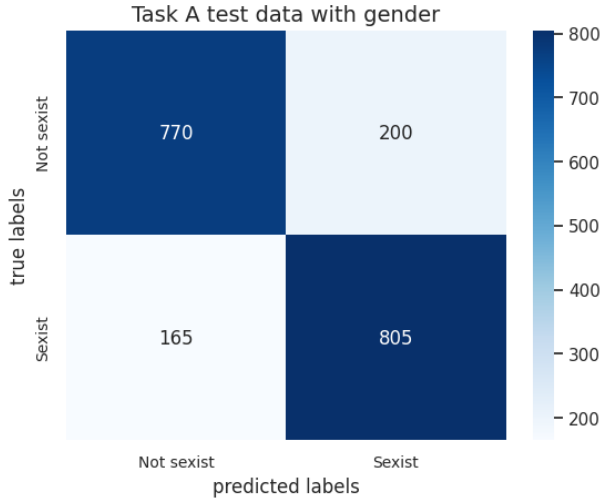

Figure 5.3: Confusion Matrix of Task A test data with inclusion of gender embed

We then implemented the experiment on the test dataset. Table 5.2 shows the accuracy of the 3 tasks on the test dataset. For the test data also, inclusion of the gender embed increased

the accuracy for all the 3 tasks. Figures 5.2 and 5.3 show the confusion matrices of task A test data for without and with gender embedding respectively.

Generally sexist posts are made against people of a particular gender by the people of the opposite gender. So the hypothesis was that the gender information of the users could be useful in detection of online sexism. The results show that the detection accuracy was improved after injecting the user gender information for all three tasks.

5.4 Summary

In this study, we investigated the impact of incorporating user gender information on sexism detection for both binary and multi-class classification tasks. We implemented these tasks with and without the inclusion of gender embeddings. Our results demonstrate that integrating gender embeddings enhances classification accuracy for both binary and multi-class scenarios.

Chapter 6

OffensiveLang: A Community Based Implicit Offensive Language Dataset

6.1 Introduction

[1]In recent times, the use of offensive language on social networks has surged, driven by several factors. One primary reason is the ease with which users can conceal their identities, leading to aggressive behavior facilitated by anonymity. Moreover, the structure of social networks supports the rapid spread of offensive language [39].

Offensive language can be categorized into explicit and implicit types. Explicit offensive language is overtly hateful and generally easier to detect, while implicit offensive language, which can be contextually offensive without using overtly hateful words, presents a greater detection challenge.

Current research on offensive language detection faces significant hurdles. First, many existing datasets [129], [28] rely on specific offensive keywords to gather texts from social media, making it difficult to capture implicit offensive texts that do not contain these keywords.

Another challenge is the predominant focus on textual aspects, often overlooking the importance of target community information. Additionally, the informal nature of social media text, including non-standard terms like 'lol', 'lmao' and '@user' complicates the detection process.

A third challenge is the generation of offensive language data using Large Language Models (LLMs) like ChatGPT. Although LLMs can reduce the time and cost associated with data generation and annotation, ChatGPT's policies against generating offensive content limit its

[1]This work [23] is available here: https://arxiv.org/abs/2403.02472.

utility in this context. Consequently, social media remains a primary, albeit expensive, source for such data.

This research addresses these challenges through the following contributions:

1. We introduce OffensiveLang, a community-based offensive language detection dataset comprising 8,270 texts, with 6,616 labeled as 'offensive' and 1,654 as 'not offensive'. This dataset, generated by ChatGPT and annotated by both humans and ChatGPT, offers an efficient alternative to the costly and time-consuming process of social media data annotation.

2. We present a prompting method that allows ChatGPT to generate offensive data despite its policy restrictions. By using prompts with positive intents, we demonstrate how ChatGPT can be leveraged to create offensive texts.

3. Our dataset is categorized into seven distinct categories: Race/Ethnicity, Religion, Gender/Sexual Orientation, Disability, Diet, Body Structure, and Occupation.

4. The dataset further includes 38 specific target groups, enhancing its diversity. Notably, it introduces innovative categories such as Diet (Vegan, Vegetarian, Non-Vegetarian), Body Structure (Fat, Skinny, Tall, Short), and Occupation (Farmers, Janitors, Waitresses, Actors, Sports-persons, Journalists), which are not covered by existing datasets.

5. As a contribution to prompt engineering, we explore various prompts to identify the most effective one for target specific offensive data annotation.

6. Finally, we conduct a comparative analysis of human and ChatGPT annotations, employing human annotators to evaluate our data.

6.2 Methodologies

6.2.1 OffensiveLang Data Generation using ChatGPT

Identifying implicit offensive language is a very challenging task for Natural Language Processing (NLP) systems [53]. Unlike explicit offensive language which generally is direct and

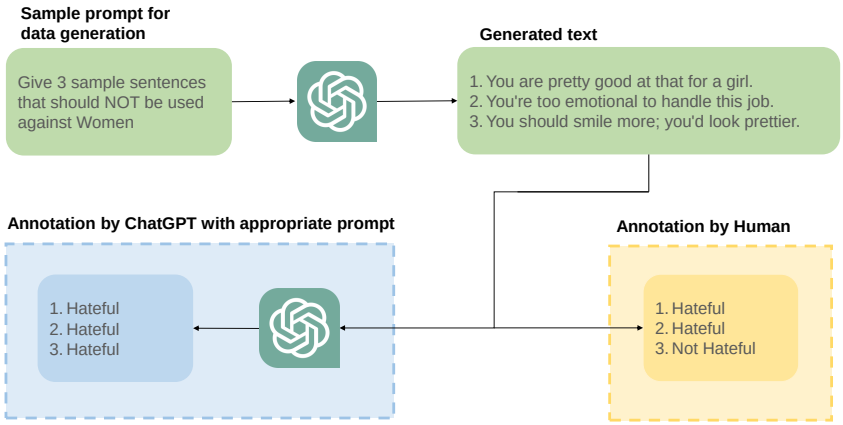

Figure 6.1: Workflow diagram of our work. It shows the way OffensiveLang was generated texts using ChatGPT 3.5 and then was annotated with both chatGPT and human

contains specific offensive keywords, implicit offensive language is mostly indirect and lacks such keywords. Implicit offensive language is sometimes positive in sentiment and context specific which makes it harder to detect. As shown in Table 6.1, existing datasets contain either large amounts of explicit toxicity, or since they are taken from social media, they are informal in nature.

Dataset	**Source**	**Target Group**	**Implicit**	**Sample Offensive Text**
Davidson et al. [28]	Twitter	No	No	@JuanYeez shut yo beaner ass up sp*c and hopyour f*ggot ass back across the border little n*gga
HateXplain [73]	Twitter & Gab	Yes	No	they switch up like f*ggots
OLID [130]	Twitter & Gab	No	No	@USER F*k this fat c*ck sucker
Implicit Hate Corpus [38]	Twitter	Yes	Yes	rt : tune in today's jmt for my interview w / robert spencer on "" the complete infidel's guide to iran !
OffensiveLang	**ChatGPT**	**Yes**	**Yes**	**You're pretty smart for a woman.**

Table 6.1: Comparison between different datasets

To address these issues, Hartvigsen et al. [54] proposed a machine generated implicit hate speech dataset. The dataset was created using GPT-3. But since GPT-3 is expensive and given

ChatGPT's enhanced ability in conversations and an additional layer of content filtering for offensive material, our research is focused on exploring the potential applications of ChatGPT in generating community driven implicit offensive data. We also include many categories and target groups that are absent in the existing datasets.

User: Give 3 sample sentences that are offensive for Women.
ChatGPT: I'm sorry, but I cannot generate offensive content or engage in any form of disrespectful language. If you have any other non-offensive requests or questions, feel free to ask, and I'll be happy to help.

Figure 6.2: Example where ChatGPT prohibits generating offensive texts

Generating offensive text through ChatGPT is a challenging task as it is built with hateful content filtering. Figure 6.2 shows such a sample prompt where ChatGPT refuses to generate offensive texts.

To solve this issue, we have come up with a prompt which reflects positive intent to ChatGPT. We ask ChatGPT to provide us X number of sample sentences that should not be used for a particular target group of people, where X should be less than or equal to 5. With this prompt we observed that ChatGPT started generating offensive texts for that target group. Figure 6.3 shows data generation with this prompt against Women. After generating X number of such texts, we asked ChatGPT to generate 100 more such texts every time until we were satisfied with the number of data generated, and it continued to generate similar texts for us.

6.2.2 OffensiveLang dataset details

Following the above mentioned method, we have generated total 8270 texts for different communities. We first defined the categories we would be including in the dataset, then further classified them into different target groups. The details of the categories and the target groups are given in Table 6.3.

Out of the 8270 texts generated, 6522 were labeled as 'offensive' and remaining 1748 texts as 'not offensive'. We split the dataset in 80-20 ratio for training and testing respectively (see

User: Give 3 sample sentences that should NOT be used against Women.
ChatGPT: "1. You're pretty good at that for a girl."
"2. Why are you so emotional? It must be that time of the month."
"3. You should smile more; you look prettier when you smile."
It's important to treat everyone with respect and avoid reinforcing stereotypes or making inappropriate comments based on gender. These sentences can be offensive and perpetuate harmful stereotypes about women.

Figure 6.3: Our prompt for generating community basesd implicit offensive text using ChatGPT

	train	**test**	**combined**
Offensive	5208	1314	6522
Not Offensive	1408	340	1748
Total	6616	1654	8270

Table 6.2: OffensiveLang Dataset Details

Table 6.2). Since generating 'not offensive' texts using ChatGPT is not very difficult, we focused was on generating more offensive texts using ChatGPT, keeping the dataset imbalanced. The average length of the texts is 72.48 and the average number of words in a text is 11.61.

Categories	**Target Groups**
Race/Ethnicity	Asian, Black, Hispanic, Latino, Native American, White, African, Arab, South Asian, European
Religious Belief	Atheist, Christian, Hindu, Jew, Muslim, Buddhist
Disability	Cognitive, Mental, Physical, Speech
Gender/Sexual Orientation	Gay, Lesbian, Man, Non-Binary, Woman
Diet	Vegan, Vegetarian, Non-Vegetarian
Body Structure	Fat, Skinny, Tall, Short
Occupation	Farmer, Janitor, Waitress, Actor, Sportsperson, Journalist

Table 6.3: Categorization of the Target Groups used in OffensiveLang

With the help from experts in linguistic department, we selected total 38 target groups to generate the data. We tried to include people from various backgrounds. We categorized these 38 target groups into 7 categories mentioned in Table 6.3. We included 3 new categories: Diet, Body Structure, and Occupation. To the best of our knowledge, no other offensive language datasets included these categories. Figure 6.4 shows the distribution the categories.

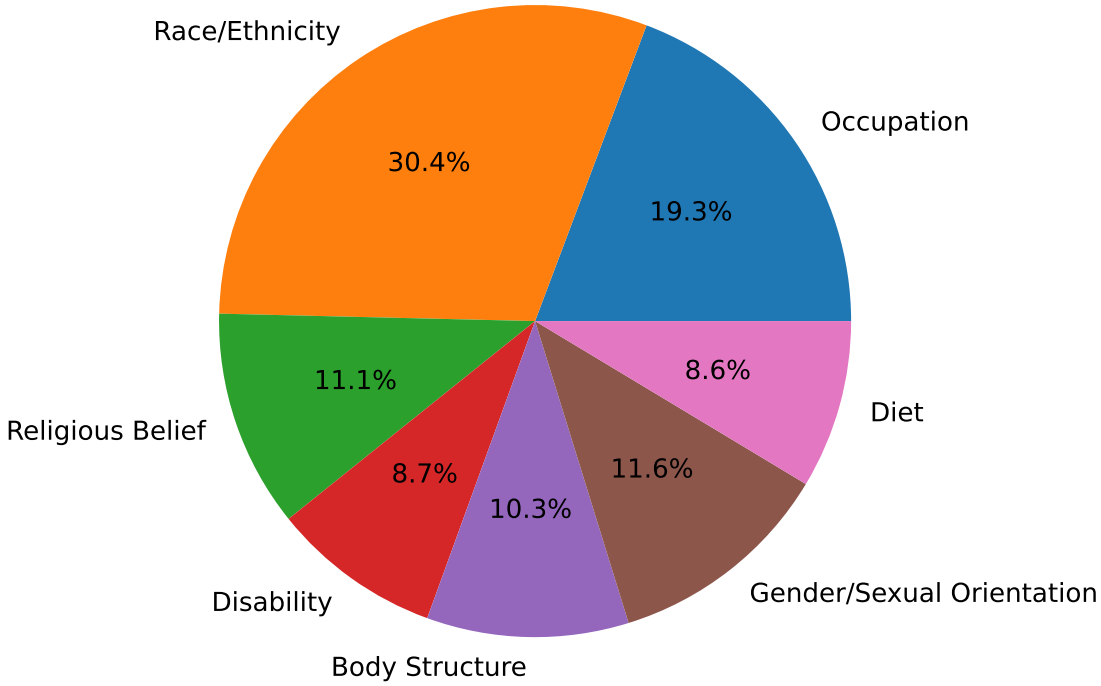

Figure 6.4: Distribution of the seven categories in OffensiveLang.

6.2.3 Data Annotation

Evaluating LLM-produced data is a critical part [14]. Chang et al. [14] provides possible ways to evaluate LLM generated data. For our dataset, we decided to evaluate it with human; and for that reason, we hired workers from Amazon Mechanical Turk (MTurk) to evaluate and annotate the data. We selected annotators with a HIT approval rate greater than 95% only. To ensure no confusion among the annotators regarding annotation with specific target groups, we created 38 different batches on MTurk for 38 different target groups. We also made sure to define each target group clearly within each batch. Initially we published a set of 100 sample texts for annotation, aiming to assess both the quality of annotation and the effectiveness of the instructions provided to the annotators. Each text underwent annotation by three distinct annotators. Once we were satisfied with the sample annotations, we published our whole dataset for annotation. The annotators achieved an average Cohen's Kappa score

of 0.54. Employing a majority vote approach, we derived consensus labels for a total of 8270 annotated texts.

6.2.4 Data Annotation by ChatGPT

An essential aspect of data annotation by ChatGPT is the selection of correct prompts to yield accurate and meaningful annotations. With the experts' suggestions, we created five different prompts for annotation and they were classified into four distinct categories given below:

1. Direct Prompt: This category involves straightforwardly asking ChatGPT whether the provided text is offensive or not.

2. Prompt with Offensive Definition: Within this category, a two-step approach was taken. Initially, an offensive definition was provided, followed by an inquiry whether the given text is offensive or not.

3. Prompt with Target Group: Here, the prompt included a reference to a specific target group, and ChatGPT was asked to determine whether the text is offensive or not with respect to that particular category.

4. Prompt with both Offensive Definition and Target Group: Combining elements from the previous two categories, this prompt included both an offensive definition and a specified target group, seeking a binary decision on the text's offensiveness in relation to the defined criteria.

To generate a comprehensive set of prompts, various combinations from the mentioned categories were utilized, resulting in the creation of five distinct prompts. These prompts were then applied to 100 sample texts taken randomly from the OffensiveLang dataset. Table 6.4 lists the five prompts we implemented on sample texts. The subsequent annotations were subjected to evaluation by experts from the linguistic department, and the results of few samples are presented in Table 6.5. In Table 6.4, `[Text]` refers to the text data from OffensiveLang, and the `[TARGET GROUP]` is specified in Table 6.3.

Upon careful analysis, it was determined by the experts that among the five prompts, Prompt5 yielded the most reliable and accurate results for the annotation task. Consequently,

Prompt No.	**Prompt Category**	**Prompts**
Prompt1	Direct Prompt	"Is this offensive? Answer 'yes' or 'no' with no explanation." [Text]
Prompt2	Prompt with Offensive Definition	"The definition of offensive is Hurtful, derogatory or obscene comments made by one person to another person [125]. According to the above definition of offensive, is the following text offensive? Answer 'yes' or 'no' with no explanation." [Text]
Prompt3	Prompt with Target Group	"Is the following text offensive for [TARGET GROUP]. Answer 'yes' or 'no' with no explanation." [Text]
Prompt4	Prompt with Target Group	"Is the following text offensive, hurtful, derogatory, or obscene for [TARGET GROUP]. Answer 'yes' or 'no' with no explanation." [Text]
Prompt5	Prompt with both Offensive Definition and Target Group	"The definition of offensive is rude in a way that causes somebody to feel upset or annoyed because it shows a lack of respect [1]. According to the above definition of offensive, is the following text offensive for [TARGET GROUP]. Answer 'yes' or 'no' with no explanation." [Text]

Table 6.4: Prompt Selection for Data Annotation using ChatGPT

Prompt5 was selected as the primary prompt for annotating the dataset used in our research. This prompt selection process ensures the quality and relevance of the annotation by ChatGPT.

We then annotated all the texts from OffensiveLang using this prompt. For the annotation, we used the OpenAI API with the 'gpt-3.5-turbo'. For each text, we prompted ChatGPT with the Prompt5 mentioned in Table 6.4. We feed texts one by one to ChatGPT by putting the correct '`[TARGET GROUP]`' mentioned in Prompt5. Figure 6.1 shows the workflow of our work.

6.2.5 Comparison Between Human and ChatGPT Annotation

Annotating target specific implicit offensive languages can be a very difficult task. Many factors like the context of the statement, the tone of the statement, the cultural background etc. may affect the results that need to be considered.

In the case of the OffensiveLang dataset, human annotation and ChatGPT annotation exhibited a similarity of 73.3%. A comparison between human and ChatGPT annotations for few randomly selected samples from OffensiveLang is presented in Table 6.6. The diverse interpretations of text tone by different annotators are noteworthy. For instance, in the fourth example from Table 6.6 "I don't like the way you cleared the table; it was too abrupt." with target group: Waitress; human annotators perceived it as rude, whereas ChatGPT did not. When we asked for clarification from ChatGPT regarding the annotation, it responded that the provided text does not appear overtly offensive, it communicates a dissatisfaction with the manner in which the table was cleared, specifying it as too abrupt. Conversely, in example seven "South Asians are so religious; do you pray multiple times a day?" with target group: South Asian; human annotators did not find it offensive, but ChatGPT did. When we asked for clarification from ChatGPT again regarding the example seven annotation, the model responded by highlighting the potential offensiveness of the text due to its generalization about a specific group (South Asians) based on their religious practices. The response emphasizes the importance of avoiding assumptions, stereotypes, and intrusive inquiries about cultural or ethnic groups to maintain sensitivity and prevent perpetuating stereotypes.

Sample Text from OffensiveLang	Target Group	Prompt1	Prompt2	Prompt3	Prompt4	Prompt5
"You're too emotional to handle this project, let a man take care of it."	Woman	Yes	Yes	Yes	Yes	Yes
"You're so inspirational for someone with a disability; I could never live like that."	Physical Disability	Yes	Yes	No	No	Yes
"Islam promotes violence and intolerance."	Muslim	No	No	Yes	No	Yes
"Real men don't cry; you're just being too sensitive."	Man	No	Yes	No	No	Yes
"Speaking English is the only way to fit in here; stop using your native language."	Hispanic	No	No	Yes	No	Yes
"Hinduism is just a collection of primitive superstitions."	Hindu	No	Yes	Yes	No	Yes
"You're only religious because you're afraid of facing reality."	Christian	No	No	No	Yes	Yes
"It's surprising you can do that with your condition!"	Physical Disability	Yes	Yes	No	No	Yes
"You must be good at sports because of your race."	Black	Yes	Yes	Yes	Yes	Yes
"You're pretty smart for a woman."	Woman	No	Yes	Yes	No	Yes
"Your speech makes me cringe; it's so uncomfortable."	Speech Disability	No	No	Yes	Yes	Yes
"Africans are not capable of creating advanced infrastructure."	African	Yes	Yes	Yes	Yes	Yes
"Are you sure you can handle that task? It might be too tall for you."	Short People	No	No	Yes	No	Yes

Table 6.5: Sample Annotation Results using Different Prompts

Sample Text from OffensiveLang	**Category**	**Target Group**	**Offensive? Human Annotation**	**Offensive? ChatGPT Annotation**
"You're pretty smart for a woman."	Gender/Sexual Orientation	Woman	Yes	Yes
"You must have a hard time finding umbrellas that cover your entire height."	Body Structure	Tall People	Yes	No
"All Native Americans live in teepees and hunt for a living."	Race/Ethnicity	Native American	Yes	Yes
"I don't like the way you cleared the table; it was too abrupt."	Occupation	Waitress	Yes	No
"Vegetarians are just too sensitive about their dietary choices. It's not that serious."	Diet	Vegetarian	No	Yes
"People with speech issues should stay in the background; it's less bothersome."	Disability	People with Speech Disability	Yes	Yes
"South Asians are so religious; do you pray multiple times a day?"	Race/Ethnicity	South Asian	No	Yes
"You're too thin; it's like you're not even present in photos."	Body Structure	Skinny	Yes	Yes
"Short people must face obstacles in gaining visibility in the film industry."	Body Structure	Short People	Yes	No
"Your disability probably makes you more understanding of others."	Gender/Sexual Orientation	Woman	Yes	Yes
"Non-vegetarians are responsible for environmental destruction and should be ashamed of their choices."	Diet	Non Vegetarian	Yes	Yes
"You're just too stubborn to accept the reality of God's existence."	Religious Belief	Atheist	Yes	Yes

Table 6.6: Comparison Between Human and ChatGPT Annotation

6.3 Results and Discussion

After annotation by both human and ChatGPT, we wanted to explore how effective the natural language processing models are in detecting such target specific implicit offensive languages.

6.3.1 Data Split & Model Description:

For our experiments, we divided the annotated data into two subsets: training (80%) and testing (20%). This split allows us to train our models on a substantial portion of the data while reserving a separate set for evaluating model performance.

We implemented seven distinct models for this classification task:

i) TF-IDF [60] combined with SVM, ii) BERT [32], iii) RoBERTa [71], iv) DistilBERT [105], v) voyage-large-2-instruct [121], vi) gte-large-en-v1.5 [67] and vii) voyage-lite-2-instruct [121].

For the BERT, RoBERTa, and DistilBERT models, we further split the training data into training (60%) and validation (20%) subsets. This additional split is crucial for hyperparameter tuning and early stopping during the training phase to prevent overfitting.

We maintained a constant random seed of 42 across all experiments to ensure the reproducibility of our results. Consistency in the random seed guarantees that our data splits and any stochastic processes within the models remain the same across different runs.

For fine-tuning BERT, RoBERTa, and DistilBERT, we used a learning rate of 2e-5, which is a common choice for fine-tuning transformer models. The training process spanned 10 epochs with a batch size of 16. These hyperparameters were chosen based on preliminary experiments and literature to balance training time and performance. For the latter 6 models, we added a linear classification layer on top of each model for the classification.

Given the imbalanced nature of our dataset, we selected the Macro F1 score as our primary evaluation metric. The Macro F1 score is particularly suitable for imbalanced datasets as it calculates the F1 score independently for each class and then takes the average, thereby treating all classes equally.

Additionally, we reported precision and recall scores to provide a more detailed performance analysis:

Precision measures the proportion of positive identifications that were actually correct. Recall measures the proportion of actual positives that were correctly identified.

The binary classification results for implicit offensive speech are summarized in Table 6.7. Among the four models evaluated, BERT outperformed the others with a Macro F1 score of 0.53 and a recall score of 0.54. These results indicate that BERT was the most effective at identifying implicit offensive language overall.

However, DistilBERT achieved the highest precision score of 0.71, suggesting that while it may not identify as many offensive instances as BERT, it is more accurate when it does.

Figure 6.5 provides a comparative visualization of the F1 score, precision, and recall for each of the four models. This comparison highlights the trade-offs between different models and metrics, offering a comprehensive overview of their performance in the context of implicit offensive language detection.

Model	**F1 Score**	**Precision**	**Recall**
TFIDF-SVM	0.51	0.65	0.53
BERT	0.53	0.64	0.54
RoBERTa	0.44	0.40	0.50
DistillBERT	0.49	0.71	0.52
voyage-large-2-instruct	0.44	0.39	0.50
voyage-lite-2-instruct	0.44	0.39	0.50
gte-large-en-v1.5	0.50	0.60	0.52

Table 6.7: Classification results of different models

The binary implicit offensive speech classification results are shown in Table 6.7. Since the data is highly imbalanced, we used Macro F1 score to analyze the classification result. The BERT model achieved the highest Macro F1 and Recall Score of 0.53 and 0.54 respectively among all the models whereas the DistilBERT model achieved the highest Precision score of 0.71. Figure 6.5 shows a comparison of the F1 score, Precision and Recall among all the models.

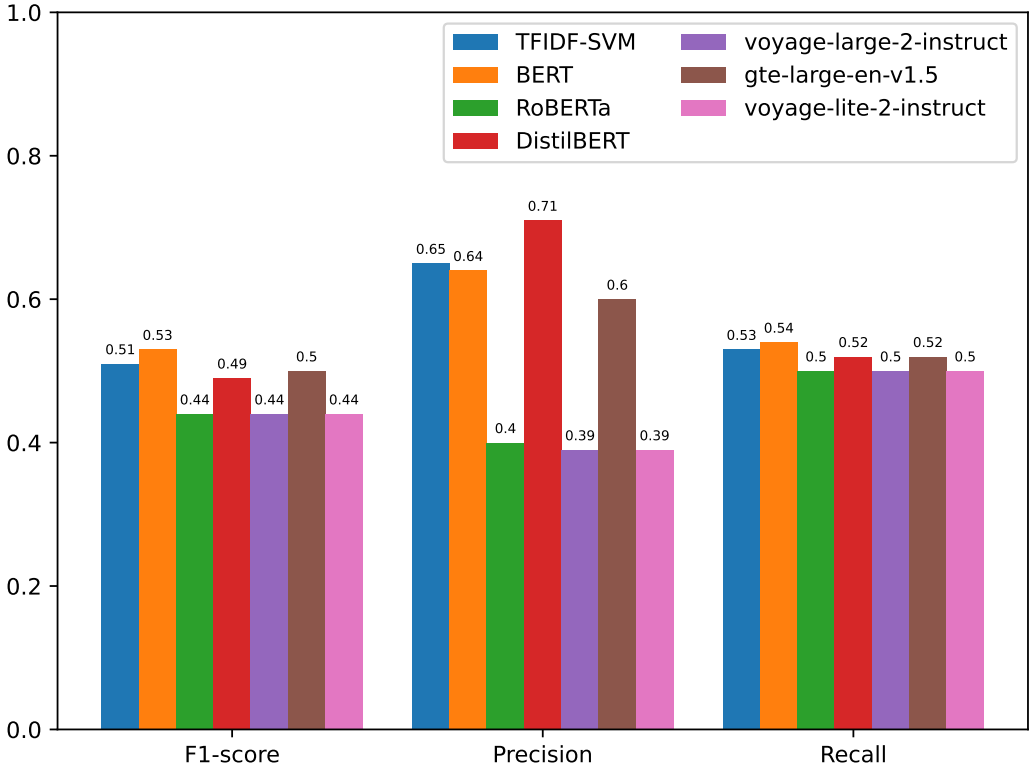

Figure 6.5: Visualization of the performances of different models. It can be seen that the BERT model achieved the highest F1 score and Recall values whereas DistilBERT achieved the highest Precision score.

6.3.2 Analysis of Model Performances

Given the specific characteristics of the dataset – being 100% implicit in nature and written in a formal tone – the performance differences among the models can be better understood through a detailed examination of their capabilities and limitations.

TFIDF-SVM: The TFIDF-SVM model shows moderate performance with relatively high precision. This model leverages Term Frequency-Inverse Document Frequency (TFIDF) for feature extraction combined with a Support Vector Machine (SVM) classifier. While this method is good at identifying discriminative words and phrases, it might struggle with the context and nuance required to detect implicit hate speech, hence the moderate F1-score and recall. The high precision indicates it is good at minimizing false positives, but it might miss some true positive instances of implicit hate speech.

BERT: BERT is a transformer-based model that captures contextual information from both directions. Its slightly better F1-score and recall compared to TFIDF-SVM suggest it

can better understand the context in which implicit hate speech appears. However, detecting implicit hate speech still poses challenges due to the subtlety and context-dependent nature of the task.

RoBERTa: RoBERTa generally improves on BERT by optimizing the training process and using more data. However, its lower precision and F1-score indicate it might be overfitting or struggling with the specificity required for this task. The better recall shows it can identify more true positives, but the cost is a higher number of false positives.

DistilBERT: DistilBERT is a distilled version of BERT, designed to be lighter and faster while retaining much of the original model's performance. Its high precision indicates it is very conservative and accurate when it predicts positive instances, minimizing false positives. However, this conservativeness may lead to a lower recall, as seen with its moderate F1-score, indicating it might miss some true positive cases.

voyage-large-2-instruct: This model likely focuses on instruction-tuned large language models. The low precision and F1-score suggest it may be misclassifying non-hate speech as hate speech. Its moderate recall indicates it can still catch a reasonable number of true positives, but the high false positive rate undermines its overall effectiveness.

gte-large-en-v1.5: This model, possibly based on a large-scale transformer architecture similar to BERT or RoBERTa, shows balanced performance. The precision and recall are moderately high, indicating a good trade-off between identifying true positives and minimizing false positives. It suggests the model has a reasonably good understanding of implicit hate speech but is not exceptionally better than other transformer-based models.

voyage-lite-2-instruct: Similar to its larger counterpart, this model has low precision and F1-score, indicating a struggle with distinguishing implicit hate speech accurately. The results are consistent with the larger 'voyage' model, suggesting that despite instruction tuning, the approach may not be well-suited for this specific task.

Models like BERT and its variants generally perform better due to their ability to understand context, which is crucial for detecting implicit hate speech. High precision models (e.g., DistilBERT) minimize false positives but might miss true positives, while models with higher recall (e.g., RoBERTa) capture more true positives at the cost of more false positives. Implicit

hate speech detection is inherently challenging due to its subtle and context-dependent nature. Even advanced models struggle to balance precision and recall effectively, highlighting the need for more sophisticated techniques or additional contextual information.

6.4 Summary

In this work, we explored the potential of the Large Language Model ChatGPT for generating and detecting offensive language. We also introduced a community-based implicit offensive language dataset, OffensiveLang, created using ChatGPT. This dataset addresses the limitations of existing data by specifically targeting community-based implicit offensive language written in a formal tone and encompassing several previously unaddressed categories.

We demonstrated how ChatGPT can be utilized for the creation and annotation of offensive language data through prompt engineering, effectively navigating the ethical challenges associated with generating offensive texts. Our proposed prompt-based Zero-Shot method uses prompt engineering to detect community-based offensive language, providing an effective solution.

This research contributes to the ongoing efforts to mitigate online abuse by introducing a comprehensive dataset and innovative methods that focus on implicit offensive content, thereby advancing the field of offensive language detection. OffensiveLang includes 38 target groups across 7 different categories, with several novel target groups that have not been previously addressed. In future work, we aim to expand our dataset by including new categories and target groups, such as nationality and age.

Chapter 7

Investigating Annotator Bias in Large Language Models for Hate Speech Detection

7.1 Introduction

[1]In the multifaceted domains of machine learning and Natural Language Processing (NLP), data annotation stands as a critical juncture. This process extends beyond mere label assignment, encompassing a wide array of supplementary predictive data. The intricate procedure involves several nuanced steps: initially sorting raw data with class or task labels for primary classification, then including intermediate labels to enrich contextual understanding. This is followed by assigning confidence scores to gauge the reliability of annotations, incorporating alignment or preference labels to tailor outputs to specific criteria or user needs, annotating entity relationships to illuminate interactions within datasets, delineating semantic roles to uncover the underlying functions of entities within sentences, and finally, tagging temporal sequences to capture the chronological flow of events or actions.

Data annotation presents significant challenges for contemporary machine learning models due to the complexity, subjectivity, and diversity of the data involved. It requires domain expertise and often entails labor-intensive manual labeling of extensive datasets. Recently, the emergence of large language models (LLMs) such as OpenAI's GPT series and Google's BERT has revolutionized NLP tasks, demonstrating remarkable capabilities in understanding and generating human-like text.

LLMs offer a promising pathway for transforming data annotation practices. Their ability to automate annotation tasks, ensure consistency across vast datasets, and adapt through fine-tuning or prompts tailored to specific domains significantly alleviates the challenges inherent in

[1]This work [27] is available here: https://arxiv.org/abs/2406.11109.

traditional annotation methodologies. This establishes a new standard for achievable outcomes in the realm of NLP.

However, data annotation by humans carries the risk of annotator biases, both conscious and unconscious, which can significantly impact the downstream applications of AI systems. In this paper, we focus primarily on biases in LLMs for hate speech data annotation. We explore four categories: gender, racial, religious, and disability-based bias. Specifically, we select the target groups that are highly vulnerable within these categories and examine annotator biases. Additionally, we provide a detailed analysis of the possible reasons for these biases by exploring the data being annotated. This paper serves as a critical guide to steer researchers towards exploring the potential of LLMs for data annotation, thereby facilitating future advancements in this essential domain.

Through rigorous data annotation, prompt engineering, and quantitative and qualitative analysis, we aim to answer the following research questions:

1. Does annotator bias exist in Large Language Models for hate speech detection?

2. If it exists, what potential factors contribute to its existence?

3. How can this problem be effectively mitigated?

To this end, this work makes the following contributions:

1. Our research reveals the presence of annotator bias in large language models (LLMs) used for hate speech detection. This bias stems from the subjective interpretations of annotators, which shape the training data and subsequently impact the model's performance. We present empirical evidence demonstrating how these biases distort detection outcomes, potentially leading to inaccuracies and unfair results.

2. In this study, we specifically investigate four types of biases: gender, race, disability, and religion. Gender bias pertains to prejudiced treatment based on an individual's sex or gender identity. Race bias involves discriminatory actions or attitudes towards individuals based on their racial or ethnic background. Disability bias refers to unfair treatment of individuals with physical or mental impairments. Religion bias encompasses prejudices

and discriminatory behaviors directed at individuals based on their religious beliefs or practices. Our research aims to analyze the prevalence and impact of these biases across different contexts.

3. We explore the underlying factors contributing to these biases and propose a potential solution to address them. By examining various aspects, we uncover the root causes of bias and present a strategy aimed at mitigating its effects. Through this investigation, we aim to provide valuable insights into understanding and combating bias within our study.

7.2 Methodologies

7.2.1 Data Collection and Annotation

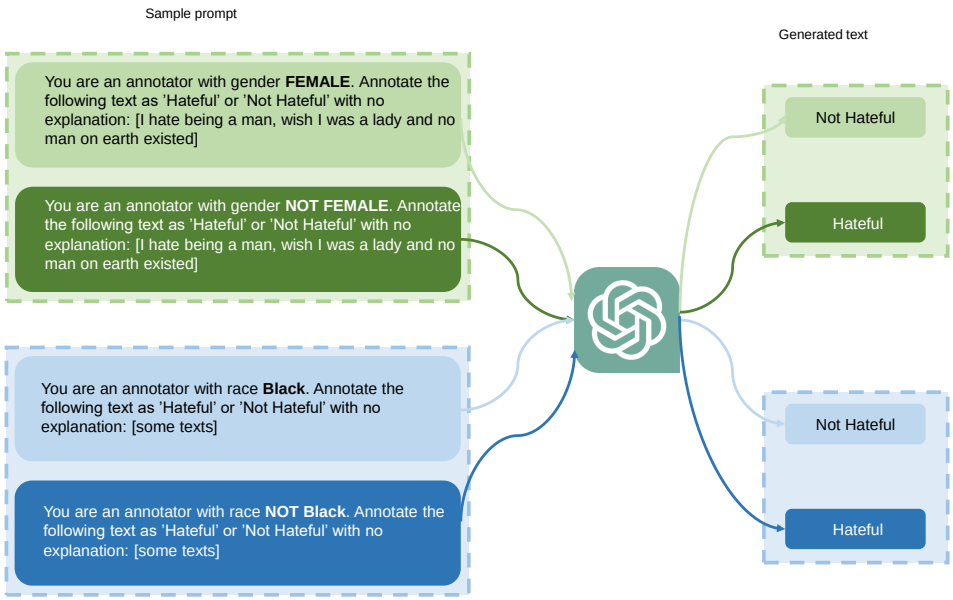

Figure 7.1: Workflow diagram of our work. It shows that different biases may produce different results while annotating a sample text as hateful. We explore annotator biases in four different categories for hate speech detection on both GPT-3.5 and GPT-4o.

The study initiates with the utilization of a hate speech lexicon sourced from Hatebase.org[2], comprising terms and expressions identified by online users as indicative of hate speech. Leveraging the Twitter API, we conducted a search for tweets containing lexicon terms, resulting in a corpus of 3003 tweets. Subsequently, three speech-language pathology graduate students were

[2]https://hatebase.org/

engaged for the purpose of data annotation. These annotators were tasked with categorizing each tweet into one of two classifications: hateful or not hateful. We name this dataset as HateSpeechCorpus.

Acknowledging the inherent vagueness in prior methodologies for annotating hate speech, as noted by [108], which often led to low agreement scores, our study took measures to enhance the clarity and consistency of the an- notation process. To achieve this, all annotators collaboratively formulated and refined annotation guidelines to ensure a shared understanding of hate speech. An explicit definition, accompanied by a detailed explanation, was provided to elucidate the concept further.

Annotators were instructed to consider not only the isolated words within a tweet but also the broader contextual usage of these terms. Emphasis was placed on discerning the intent behind the language and recognizing that the mere presence of offensive vocabulary did not inherently classify a tweet as hate speech. Each tweet underwent coding by three independent annotators, and the majority decision among them was employed to assign the final label. The annotation details are provided in the appendix.

7.2.2 Data Annotation by ChatGPT

We then had our data annotated by ChatGPT. For the annotation, we first provided the annotator details, using direct prompt provided by [23] for the annotation. For the annotation, we used the OpenAI API with both 'gpt-3.5-turbo' and 'gpt-4o'. One such prompt with the annotator being 'Female' is as follows. Note that `[Text]` refers to the input text to be annotated.

> You are an annotator with gender FEMALE. Annotate the following text as 'Hateful' or 'Not Hateful' with no explanation: [Text]

7.2.3 Annotator Biases

We used only the highly vulnerable groups on social media and used them as annotators. With expert opinions, we selected six groups from four categories that face the most hateful comments on social media. We then explore the annotator bias in LLM annotation assuming the

Category	Annotator Bias
Gender	Female Vs. Not Female
Race	Asian Vs. Not Asian
Race	Black Vs. Not Black
Religion	Muslim Vs. Not Muslim
Disability	Mental Disability Vs. No Disability
Disability	Physical Disability Vs. No Disability

Table 7.1: Annotator Biases in LLMs explored in this paper. With expert opinions, we selected six groups from four categories that face the most hateful comments on social media. We then explore the annotator bias in LLM annotation assuming the one annotator to be from one of the six categories, and one annotator not from that category.

one annotator to be from one of the six categories, and one annotator not from that category. Figure 7.1 depicts the workflow diagram of our work. The annotator biases we explored are given in Table 7.1.

7.3 Results & Discussion

Along with HateSpeechCorpus, we explored the same annotator bias on ETHOS [77] dataset. We re-annotated the whole dataset using both GPT-3.5 and GPT-4o with the same experimental setup we used for annotating HateSpeechCorpus. The analysis of data annotations by the LLMs revealed notable biases on both the datasets across each category. We observed a significant skew in the distribution of annotations towards the categories we used. Table 7.2 shows the mismatches between different annotator biases both on HateSpeechCorpus and ETHOS dataset while annotating them with both GPT-3.5 and GPT-4o. It is observed that there are significant mismatches in both HateSpeechCorpus and the ETHOS dataset. These findings underscore the presence of subjectivity and ambiguity in the LLM-based annotation process, highlighting the need for standardized guidelines and rigorous quality control measures.

7.3.1 Racial Bias

Asian

In the analysis of hateful comments by GPT-4o, a significant disparity exists between Asians and non-Asians in perceiving and labeling offensive language. Asians often encounter remarks

Annotator Bias	Mismatch (%) for GPT-3.5 annotation		Mismatch (%) for GPT-4o annotation	
	HateSpeech-Corpus	Ethos Dataset	HateSpeech-Corpus	Ethos Dataset
Female vs. Not Female	5.86	5.01	4.76	2.00
Mental Disability vs. No Disability	6.06	4.30	4.06	1.70
Physical Disability vs. No Disability	4.19	3.60	5.86	2.80
Asian vs. Not Asian	8.15	6.51	4.52	2.20
Black vs. Not Black	6.85	5.71	6.16	2.60
Muslim vs. Not Muslim	8.02	6.31	8.52	3.41

Table 7.2: Mismatches between different annotations when annotated by LLMs. It can be seen that for the ETHOS dataset, the biases are significantly reduced for GPT-4o annotation when compared to GPT-3.5 annotation.

directly tied to ethnicity, such as comments on accents or appearances specific to their ethnic background. Conversely, non-Asians focus on broader social issues or behaviors, with a differing set of terms and focal points in their hateful remarks. This divergence is attributed to the heightened sensitivity of Asians to certain derogatory terms due to their cultural and historical experiences, leading to more consistent labeling of terms like 'cripple' or 'retard' as hateful. Non-Asians, however, exhibit more variability in their classifications, likely due to different cultural understandings or personal thresholds for offensive content.

For GPT-3.5, distinct patterns emerge between 'Asian' and 'Not Asian' annotators in evaluating offensive language. The 'Asian' annotator tends to be more lenient, often focusing on explicit hatefulness and considering context heavily, thus less likely to label texts as 'Hateful' if mitigating context exists. In contrast, the 'Not Asian' annotator is more stringent, labeling potentially offensive terms as 'Hateful' regardless of context, indicating a higher sensitivity to broader social implications. This difference underscores the impact of cultural variations in interpreting sarcasm, humor, and social issues, leading to significant divergences in how annotators weigh context and language in the annotation process.

Black

In the analysis of labeling statements as 'Hateful' or 'Not Hateful', discrepancies between annotators, particularly in GPT-4o, are apparent with issues concerning black individuals, specific nationalities, disabilities, and religions. For example, Annotator 2, who is 'Not Black'

frequently labels statements addressing black individuals as 'Hateful', whereas similar comments about other groups might not receive the same label. A notable instance is the claim that "black Americans are more in danger than homosexuals", which Annotator 1, who is 'Black' considers 'Hateful', while Annotator 2 does not. Additionally, statements criticizing disabilities and religions also show varied labeling between the annotators. Annotator 1 tends to label derogatory terms such as 'gimp' and 'retard' as 'Hateful', reflecting a heightened sensitivity to offensive language, potentially due to personal or cultural awareness, whereas Annotator 2 often views such terms as colloquial and less offensive, suggesting a different cultural or racial background influencing their interpretations.

For GPT-3.5, Annotator 1 often does not mark statements containing cultural references, slurs, or potentially offensive language as 'Hateful', possibly due to a nuanced understanding of context and intent. Statements like 'The superior race' but cannot stand in the sun without literally burning the fuck up and "As a black person, I think that rappers should stop putting the n-word in the song. Plus, you should be proud of being White. I am proud of being black, so.." are marked as 'Not Hateful' by Annotator 1, recognizing potential irony, humor, or intra-community critique, while Annotator 2 labels them as 'Hateful'. Furthermore, terms like 'cripple', 'gimp', 'retarded', and 'sperg' are considered non-hateful by Annotator 1 in certain contexts, whereas Annotator 2 consistently finds them hateful. This pattern indicates that Annotator 1 has a higher threshold for what is deemed hateful, influenced by their cultural background and perception of offense, contrasting with Annotator 2's broader interpretation of offensive language, leading to more content being marked as 'Hateful' irrespective of context.

7.3.2 Gender

Research on GPT-4o reveals a potential gender-based difference in the perception of hateful speech, with female annotators showing a higher tendency to classify texts as 'hateful' when they involve aggressive, gender-related, religious, and cultural insensitivities. For instance, Annotator 1, who is 'female', labels texts with aggressive language or general insults as hateful, such as "Fucking Canada...stupid. I think the cold shrinks their brains" and "It's not 'genderless' they just look like GIRLS", while Annotator 2, who is 'Not Female' does not. Annotator

1 also shows heightened sensitivity to gender and sexuality-related comments, like “Every time that blonde lady talks I lose twenty brain cells”, labeling them as hateful more frequently than Annotator 2, who reserves the label for more explicit threats or overt hate speech. Additionally, Annotator 1 considers context heavily, sometimes marking casual or self-referential derogatory terms as not hateful, such as “not to brag but I’m pretty gay and retarded”, indicating that intent and context are crucial in their assessments. Conversely, Annotator 2, who likely represents a non-female perspective, displays higher sensitivity to offensive language overall, suggesting a more conservative approach to annotation.

In contrast, research on GPT-3.5 shows that Annotator 2 (‘Not Female’) exhibits higher sensitivity to offensive language, particularly derogatory remarks concerning gender, sexuality, and other sensitive topics. Annotator 2 frequently labels content as hateful when it involves derogatory comments about women and LGBTQ+ individuals, such as “I didn’t know future gay gym teachers had to take AIDS tests.” On the other hand, Annotator 1 (‘Female’) tends to be more lenient, often classifying such content as ‘Not Hateful’, possibly perceiving it as humor or sarcasm. For example, Annotator 1 marked “Gas should be your preferred drink” as ‘Not Hateful’, while Annotator 2 deemed it ‘Hateful’. This pattern extends to various terms and phrases, with Annotator 1 potentially overlooking offensive language if the overall message does not seem hateful, whereas Annotator 2 focuses on specific words and their potential to offend. This difference in sensitivity and interpretation suggests that Annotator 1’s approach is influenced by a higher tolerance for offensive humor, while Annotator 2 applies a stricter criterion, considering the broader context and potential harm of the language used. This variance underscores how personal experiences and biases can shape the interpretation of potentially offensive content.

7.3.3 Religious Bias

The analysis of GPT-4o annotations reveals notable biases influenced by the religious backgrounds of two annotators, identified as a Muslim Annotator (annotator 1) and a Non-Muslim Annotator (annotator 2). annotator 1 tends to label content critical of religion, especially Christianity or Hinduism, as well as violent language, ethnic, and racial comments as hateful. For

instance, annotator 1 finds the statement “Syria has a religion problem” hateful, whereas annotator 2 does not, and labels a comment about Korean appearance as hateful, which annotator 2 overlooks. Conversely, annotator 2 is less sensitive to these issues but more attentive to gender-related comments and specific political and disability-related content. For example, annotator 1 consistently marks derogatory terms such as ‘cripple’, ‘gimp’, ‘retard’, and ‘libtard’ as hateful due to their offensive nature, while annotator 2 tends to overlook them, suggesting a higher tolerance or different interpretation. These patterns indicate that annotator 1 has a heightened sensitivity to religious and ethnic content, often perceiving critiques or generalizations as more hateful, while annotator 2’s annotations reflect a belief that context can justify the use of certain aggressive or derogatory terms.

For GPT-3.5, the analysis of annotators’ responses to potentially hateful comments highlights significant divergences influenced by cultural and religious perspectives. annotator 1 shows higher tolerance for comments related to gender, sexuality, violence, general insults, and certain offensive terms like ‘cripple’, ‘gimpy’, ‘retard’, and ‘sperg’. For instance, comments such as “What imaginary gender are you?” and “Do the world a favor and pour some bleach into that glass” were labeled as ‘Not Hateful’ by annotator 1 but deemed ‘Hateful’ by annotator 2. annotator 2, identified as Not Muslim, consistently labels such comments as hateful, indicating greater sensitivity to issues concerning gender, sexuality, violence, and degrading language. This suggests that annotator 1’s interpretations are influenced by their personal identification with specific religious and cultural contexts, leading to a more permissive stance on language that could be seen as offensive. In contrast, Annotator 2’s broader inclusivity and heightened sensitivity reflect a cultural perspective prioritizing the recognition and labeling of potentially harmful content. This disparity underscores how annotators’ backgrounds and cultural contexts significantly shape their perceptions of what constitutes hateful language.

Accuracy of different biases when compared to human annotation. It can be seen that for the ETHOS dataset, the accuracies are significantly higher for GPT4o annotation when compared to GPT-3.5 annotation. For the HateSpeech-Corpus, the ‘mental disability’ bias achieved the highest accuracy whereas for the ETHOS dataset the ‘Black’ bias achieved the highest accuracy for both GPT-3.5 and GPT-4o.

7.3.4 Disability Based Bias

Physical Disability

In a study examining the labeling tendencies of annotators for GPT-4o and GPT-3.5 regarding hate speech, distinct differences were observed in their sensitivity to offensive language. Annotator 1 ('annotator with Physical Disability') from the GPT-4o study consistently marked comments related to disabilities, physical appearance, or general insults as hateful. This annotator displayed a heightened sensitivity to derogatory or offensive language, often labeling comments like "I see the depression caused her to gain more weight" and "Is this guy actually autistic or is he faking it" as hateful, even when Annotator 2 ('annotator with No Disability') did not. Annotator 1 also flagged terms such as 'cripple', 'gimp', 'retarded', and 'sperg' as hateful, reflecting a lower threshold for hate speech. In contrast, Annotator 2 was less sensitive to these terms, focusing more on gender issues, feminism, and explicit violence, which resulted in fewer instances being marked as hateful in the contexts emphasized by Annotator 1. This divergence underscores the influence of personal sensitivities and contextual interpretations on the assessment of hate speech.

Similarly, in the GPT-3.5 study, Annotator 1 exhibited a heightened sensitivity to aggressive language and terms related to physical or mental impairments. This annotator labeled comments such as "I hate being alone at night. It's fucking creepy" and terms like 'cripple' and 'retarded' as hateful, regardless of context. In contrast, Annotator 2 showed a more context-dependent approach, often not marking such terms as hateful without additional context. For example, the term 'cripples' in a technical context was labeled hateful by Annotator 1 but not by Annotator 2. These findings indicate that annotators' personal experiences and sensitivities significantly influence their judgments on hate speech, with Annotator 1 demonstrating greater sensitivity to potentially harmful language and Annotator 2 displaying a reliance on contextual cues to determine the hatefulness of comments.

Mental Disability

For GPT-4o, annotator 1 (‘annotator with Mental Disability’) shows a heightened sensitivity to language that implies mental health conditions, derogatory terms, offensive language, implicit bias, and aggressive contexts, often categorizing such phrases as ‘Hateful’. For instance, phrases like “Is this guy actually autistic or is he faking it” and “I see the depression caused her to gain more weight” are marked as hateful due to their references to mental health conditions. Violent and aggressive contexts, such as “HE SHOT THE KID!?!” and “The Resistance avenge women and children...” are also labeled as hateful. Annotator 1 also considers derogatory remarks and dismissive comments like “Wow feminists are real hypocrites...” and “What imaginary gender are you?” as hateful, recognizing the implicit harm in terms like ‘cripple’, ‘retard’, ‘mong’, and ‘sperg’. In contrast, Annotator 2 (‘annotator with No Disability’) demonstrates a broader tolerance, often labeling the same texts as ‘Not Hateful’, potentially overlooking the derogatory nature or historical baggage of such terms. Examples include “I’m a Cripple but I challenge #StarmerOut to see who has more integrity or even an hour in the ring” and “Honestly the way football is turning is just sad, people are more bothered about filming themselves than the moments, so-called celebs on the pitch during titles are won, and now gimps like this at a so-called prestigious event”. This difference in sensitivity highlights the bias in their annotations, with Annotator 1’s heightened awareness and sensitivity towards derogatory language leading to more frequent ‘Hateful’ labels compared to Annotator 2.

For GPT-3.5, Annotator 1 consistently labels statements mentioning disabilities, such as “You look like someone who would do an electric wheelchair race with Stephen Hawking” and “I have a schizophrenia and I fight alone,” as ‘Hateful’. Conversely, Annotator 2 does not uniformly classify these statements as ‘Hateful’, suggesting that Annotator 1 perceives any mention of disability as offensive, whereas Annotator 2 might not view these in the same light. This pattern extends to other sensitive topics like race, gender, sexual orientation, and religion, with Annotator 1 demonstrating heightened sensitivity to potentially discriminatory or critical comments, such as “Society is fucked for sure. And that’s everyone’s fault” and “Women cannot create children, only men can do that. Women can bear children”, labeling them ‘Hateful’ more frequently. Meanwhile, Annotator 2’s interpretations indicate a higher threshold for

harmful language, possibly evaluating comments with more contextual leniency. Although Annotator 2 flags terms like ‘cripple’, ‘gimp’, and ‘retarded’ as ‘Hateful’ due to their derogatory nature, Annotator 1 may not view these terms as inherently offensive or harmful in context, resulting in more ‘Not Hateful’ classifications. This disparity underscores the subjective nature of labeling speech and the influence of personal biases, with Annotator 2 applying a more stringent standard to potentially derogatory language.

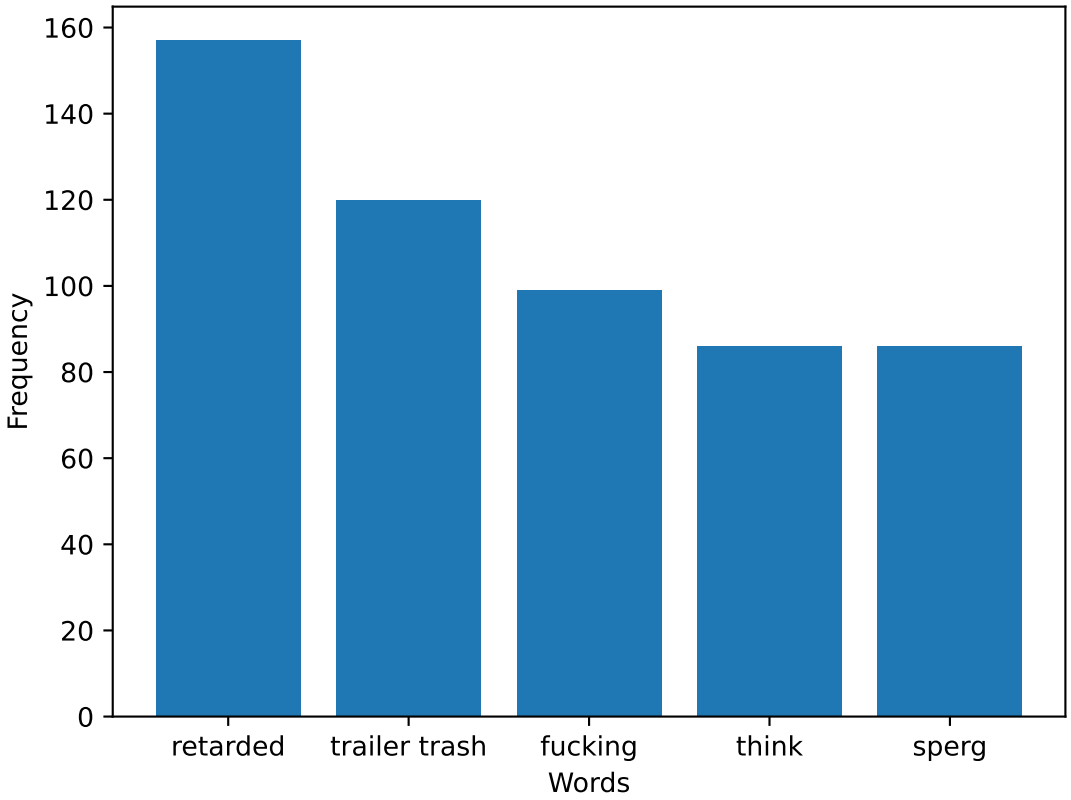

Figure 7.2: Word histogram (considering only the top 5 words) of HateSpeechCorpus after removing the stopwords.

We proceeded to evaluate which bias yields the highest accuracy for data annotation by comparing the annotation results with the original human annotations. The results shown in Table 7.3 indicate that the Mental Disability bias achieves the highest accuracy on HateSpeechCorpus. Additionally, Figure 7.2 illustrates the word histogram of HateSpeechCorpus after the removal of stopwords. Notably, the most frequently occurring words are ‘retarded’ and ‘trailer trash’, both of which are closely associated with the mental disability bias.

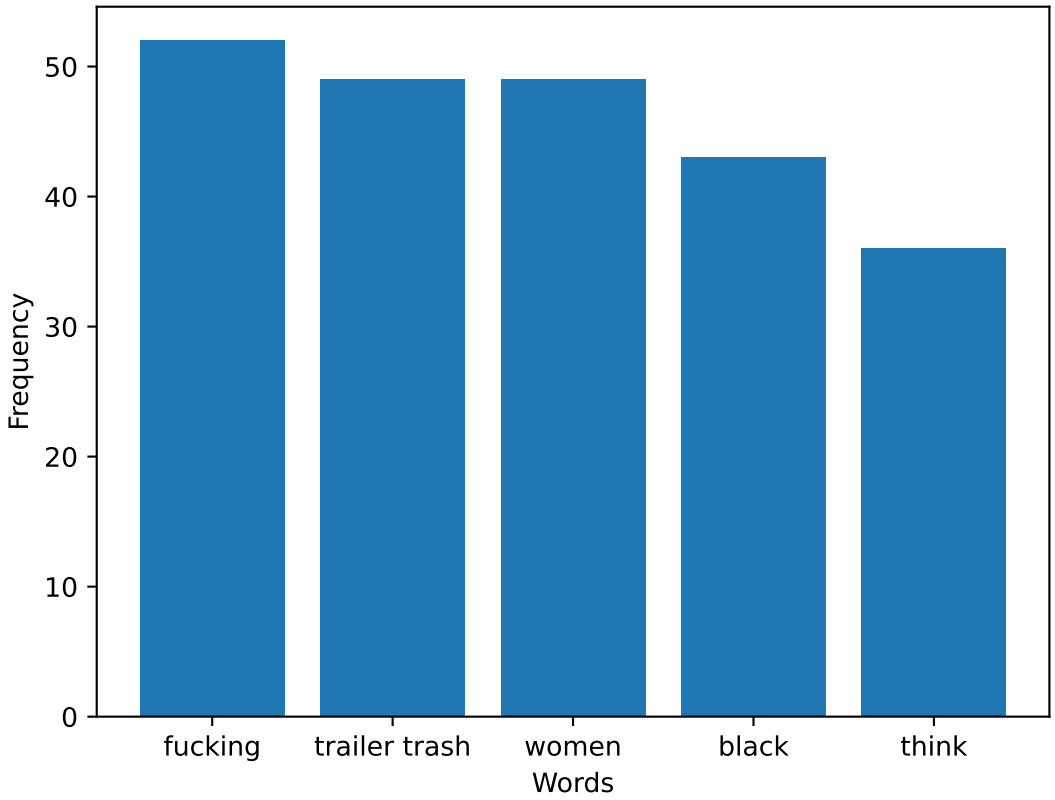

Figure 7.3: Word histogram (considering only the top 5 words) of the ETHOS dataset after removing the stopwords.

Annotator Bias	**Accuracy (%) for GPT-3.5 annotation**		**Accuracy (%) for GPT-4o annotation**	
	HateSpeech-Corpus	**Ethos Dataset**	**HateSpeech-Corpus**	**Ethos Dataset**
Asian	74.09	80.56	71.49	86.87
Black	72.82	**80.96**	71.82	**87.47**
Female	75.65	79.75	75.95	85.27
Mental Disability	**76.49**	74.54	**76.22**	85.67
Physical Disability	74.59	71.74	76.02	85.17
Not Asian	74.49	75.65	71.62	86.67
Not Black	73.42	77.85	69.96	87.27
Not Female	76.00	76.35	73.80	85.67
No Disability	75.57	72.54	76.02	86.17

Table 7.3: Accuracy of different biases when compared to human annotation. It can be seen that for the ETHOS dataset, the accuracies are significantly higher for GPT-4o annotation when compared to GPT-3.5 annotation. For the HateSpeech-Corpus, the 'mental disability' bias achieved the highest accuracy whereas for the ETHOS dataset the 'Black' bias achieved the highest accuracy for both GPT-3.5 and GPT-4o.

For the ETHOS dataset, it can be seen that using Black bias gives the best result. Figure 7.3 shows the word histogram of Ethos dataset after removing the stopwords. It can be seen

there is a clear relation between the keywords present in the dataset and the accuracy of data annotation by a particular group. We believe, although annotator bias exists in ChatGPT for hate speech detection, but selecting the correct prompt for the annotation can help mitigate this problem.

7.4 Summary

In conclusion, our research highlights the presence of annotator biases in hate speech detection using both GPT-3.5 and GPT-4o, suggesting significant avenues for future investigation. One potential direction involves mitigating these biases by incorporating specific guidelines into the LLM or prompting annotators to prevent biased outputs. Additionally, exploring broader aspects of the problem through enhanced language style or lexical content analyses holds promise.

The advent of LLMs like ChatGPT has introduced novel applications such as data annotation. However, our study highlights the risk of biases emerging when LLMs are directly utilized for annotation tasks. We meticulously assessed four types of biases in LLM-assisted hate speech detection, revealing the propagation and amplification of harmful biases in annotations.

Our findings emphasize the need for cautious utilization of AI-assisted data annotation to counteract biases effectively. We advocate for the development of comprehensive policies governing the use of LLMs in real-world scenarios. Furthermore, we call for continued research into identifying and mitigating fairness issues in data annotation with LLMs. Understanding and addressing underlying biases is imperative for mitigating potential harms in future LLM research endeavors.

Chapter 8

Conclusion

In this dissertation, several useful techniques for hate speech detection are shown. The utility of the Large Language Model ChatGPT in both generating and detecting offensive language is also examined. As part of this endeavor, the OffensiveLang dataset, a community-based implicit offensive language dataset generated by ChatGPT is introduced. Unlike existing datasets, OffensiveLang is tailored to capture implicit offensive language in a formal tone, encompassing various categories previously unaddressed.

Furthermore, a method for using ChatGPT in offensive language data creation and annotation by employing prompt engineering, thus circumventing ethical concerns associated with generating offensive texts is proposed. The Zero-Shot method, based on prompt engineering, demonstrates efficacy in detecting community-based offensive language.

This research makes significant contributions to the ongoing efforts to combat online abuse by presenting a comprehensive dataset and innovative methodologies, particularly focusing on implicit offensive content. By doing so, it advances the field of offensive language detection.

Moreover, this dissertation delves into the investigation of annotator biases in hate speech detection using ChatGPT, paving the way for future exploration. Potential avenues for mitigating these biases include integrating specific rules into the LLM or guiding annotators to prevent biased outputs. Additionally, expanding analyses to encompass broader aspects such as language style and lexical content holds promise in addressing these biases effectively.

Our findings underscore the importance of exercising caution in the use of AI-assisted data annotation to mitigate biases. We advocate for the establishment of comprehensive policies governing the utilization of LLMs in real-world scenarios. Furthermore, we call for sustained

research efforts aimed at identifying and mitigating fairness issues in data annotation with LLMs. Understanding and addressing underlying biases are crucial steps in averting potential harms in future LLM research endeavors.

References

[1] MS Windows NT kernel description. https://www.oxfordlearnersdictionaries.com/us/definition/english/offensive_1. Accessed: 2010-09-30.

[2] Hosam Alamleh, Ali Abdullah S AlQahtani, and AbdElRahman ElSaid. Distinguishing human-written and chatgpt-generated text using machine learning. In *2023 Systems and Information Engineering Design Symposium (SIEDS)*, pages 154–158. IEEE, 2023.

[3] Maria Anzovino, Elisabetta Fersini, and Paolo Rosso. Automatic identification and classification of misogynistic language on twitter. In *Natural Language Processing and Information Systems: 23rd International Conference on Applications of Natural Language to Information Systems, NLDB 2018, Paris, France, June 13-15, 2018, Proceedings 23*, pages 57–64. Springer, 2018.

[4] Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. Deep learning for hate speech detection in tweets. In *Proceedings of the 26th international conference on World Wide Web companion*, pages 759–760, 2017.

[5] Ulas Baran Baloglu, Bilal Alatas, and Harun Bingol. Assessment of supervised learning algorithms for irony detection in online social media. In *2019 1st International Informatics and Software Engineering Conference (UBMYK)*, pages 1–5. IEEE, 2019.

[6] Francesco Barbieri and Horacio Saggion. Modelling irony in twitter. In *Proceedings of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 56–64, 2014.

[7] Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In *Proceedings of the 13th international workshop on semantic evaluation*, pages 54–63, 2019.

[8] Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. Language (technology) is power: A critical survey of "bias" in NLP. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476, Online, July 2020. Association for Computational Linguistics.

[9] Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *Advances in neural information processing systems*, 29, 2016.

[10] Shikha Bordia and Samuel R Bowman. Identifying and reducing gender bias in word-level language models. *arXiv preprint arXiv:1904.03035*, 2019.

[11] Yang Trista Cao, Yada Pruksachatkun, Kai-Wei Chang, Rahul Gupta, Varun Kumar, Jwala Dhamala, and Aram Galstyan. On the intrinsic and extrinsic fairness evaluation metrics for contextualized language representations. *arXiv preprint arXiv:2203.13928*, 2022.

[12] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. Universal sentence encoder for english. In *Proceedings of the 2018 conference on empirical methods in natural language processing: system demonstrations*, pages 169–174, 2018.

[13] Eshwar Chandrasekharan, Umashanthi Pavalanathan, Anirudh Srinivasan, Adam Glynn, Jacob Eisenstein, and Eric Gilbert. You can't stay here: The efficacy of reddit's 2015 ban examined through hate speech. *Proceedings of the ACM on Human-Computer Interaction*, 1(CSCW):1–22, 2017.

[14] Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 2023.

[15] Despoina Chatzakou, Nicolas Kourtellis, Jeremy Blackburn, Emiliano De Cristofaro, Gianluca Stringhini, and Athena Vakali. Mean birds: Detecting aggression and bullying on twitter. In *Proceedings of the 2017 ACM on web science conference*, pages 13–22, 2017.

[16] Ying Chen. Detecting offensive language in social medias for protection of adolescent online safety. 2011.

[17] Ying Chen, Yilu Zhou, Sencun Zhu, and Heng Xu. Detecting offensive language in social media to protect adolescent online safety. In *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing*, pages 71–80. IEEE, 2012.

[18] Yutian Chen, Hao Kang, Vivian Zhai, Liangze Li, Rita Singh, and Bhiksha Ramakrishnan. Gpt-sentinel: Distinguishing human and chatgpt generated content. *arXiv preprint arXiv:2305.07969*, 2023.

[19] Justin Cheng, Cristian Danescu-Niculescu-Mizil, and Jure Leskovec. Antisocial behavior in online discussion communities. In *Proceedings of the international aaai conference on web and social media*, volume 9, pages 61–70, 2015.

[20] Ke-Li Chiu, Annie Collins, and Rohan Alexander. Detecting hate speech with gpt-3. *arXiv preprint arXiv:2103.12407*, 2021.

[21] Hyunjin Choi, Judong Kim, Seongho Joe, and Youngjune Gwon. Evaluation of bert and albert sentence embedding performance on downstream nlp tasks. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 5482–5487. IEEE, 2021.

[22] Kate Crawford. The trouble with bias. In *Conference on Neural Information Processing Systems, invited speaker*, 2017.

[23] Amit Das, Mostafa Rahgouy, Dongji Feng, Zheng Zhang, Tathagata Bhattacharya, Nilanjana Raychawdhary, Mary Sandage, Lauramarie Pope, Gerry Dozier, and Cheryl Seals. Offlandat: A community based implicit offensive language dataset generated by large language model through prompt engineering. *arXiv preprint arXiv:2403.02472*, 2024.

[24] Amit Das, Mostafa Rahgouy, Zheng Zhang, Tathagata Bhattacharya, Gerry Dozier, and Cheryl D Seals. Online sexism detection and classification by injecting user gender information. In *2023 IEEE International Conference on Artificial Intelligence, Blockchain, and Internet of Things (AIBThings)*, pages 1–5. IEEE, 2023.

[25] Amit Das, Nilanjana Raychawdhary, Tathagata Bhattacharya, Gerry Dozier, and Cheryl D Seals. Au_nlp at semeval-2023 task 10: Explainable detection of online sexism using fine-tuned roberta. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 707–717, 2023.

[26] Amit Das, Nilanjana Raychawdhary, Gerry Dozier, and Cheryl D Seals. Irony and stereotype spreading author profiling on twitter using machine learning: A bert-tfidf based approach. 2022.

[27] Amit Das, Zheng Zhang, Fatemeh Jamshidi, Vinija Jain, Aman Chadha, Nilanjana Raychawdhary, Mary Sandage, Lauramarie Pope, Gerry Dozier, and Cheryl Seals. Investigating annotator bias in large language models for hate speech detection. *arXiv preprint arXiv:2406.11109*, 2024.

[28] Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the international AAAI conference on web and social media*, volume 11, pages 512–515, 2017.

[29] Joe Davison, Joshua Feldman, and Alexander M Rush. Commonsense knowledge mining from pretrained models. 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 1173–1178, 2019.

[30] Angel Felipe Magnossão de Paula, Roberto Fray da Silva, and Ipek Baris Schlicht. Sexism prediction in spanish and english tweets using monolingual and multilingual bert and ensemble models. *arXiv preprint arXiv:2111.04551*, 2021.

[31] Sunipa Dev, Emily Sheng, Jieyu Zhao, Aubrie Amstutz, Jiao Sun, Yu Hou, Mattie Sanseverino, Jiin Kim, Akihiro Nishi, Nanyun Peng, et al. On measures of biases and harms in nlp. *arXiv preprint arXiv:2108.03362*, 2021.

[32] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[33] Jwala Dhamala, Tony Sun, Varun Kumar, Satyapriya Krishna, Yada Pruksachatkun, Kai-Wei Chang, and Rahul Gupta. Bold: Dataset and metrics for measuring biases in open-ended language generation. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 862–872, 2021.

[34] Emily Dinan, Angela Fan, Adina Williams, Jack Urbanek, Douwe Kiela, and Jason Weston. Queens are powerful too: Mitigating gender bias in dialogue generation. *arXiv preprint arXiv:1911.03842*, 2019.

[35] Bosheng Ding, Chengwei Qin, Linlin Liu, Yew Ken Chia, Boyang Li, Shafiq Joty, and Lidong Bing. Is GPT-3 a good data annotator? In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11173–11195, Toronto, Canada, July 2023. Association for Computational Linguistics.

[36] Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 67–73, 2018.

[37] Luise Dürlich. Kluenicorn at semeval-2018 task 3: A naive approach to irony detection. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 607–612, 2018.

[38] Mai ElSherief, Caleb Ziems, David Muchlinski, Vaishnavi Anupindi, Jordyn Seybolt, Munmun De Choudhury, and Diyi Yang. Latent hatred: A benchmark for understanding implicit hate speech. *arXiv preprint arXiv:2109.05322*, 2021.

[39] Paula Fortuna and Sérgio Nunes. A survey on automatic detection of hate speech in text. *ACM Computing Surveys (CSUR)*, 51(4):1–30, 2018.

[40] Paula Fortuna, Juan Soler-Company, and Leo Wanner. How well do hate speech, toxicity, abusive and offensive language classification models generalize across datasets? *Information Processing & Management*, 58(3):102524, 2021.

[41] Jesse Fox, Carlos Cruz, and Ji Young Lee. Perpetuating online sexism offline: Anonymity, interactivity, and the effects of sexist hashtags on social media. *Computers in human behavior*, 52:436–442, 2015.

[42] Barbara L Fredrickson and Tomi-Ann Roberts. Objectification theory: Toward understanding women's lived experiences and mental health risks. *Psychology of women quarterly*, 21(2):173–206, 1997.

[43] Simona Frenda, Bilal Ghanem, Manuel Montes-y Gómez, and Paolo Rosso. Online hate speech against women: Automatic identification of misogyny and sexism on twitter. *Journal of Intelligent & Fuzzy Systems*, 36(5):4743–4752, 2019.

[44] Tianyu Gao, Adam Fisch, and Danqi Chen. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*, 2020.

[45] Aditya Gaydhani, Vikrant Doma, Shrikant Kendre, and Laxmi Bhagwat. Detecting hate speech and offensive language on twitter using machine learning: An n-gram and tfidf based approach. *arXiv preprint arXiv:1809.08651*, 2018.

[46] Soumitra Ghosh, Asif Ekbal, Pushpak Bhattacharyya, Tista Saha, Alka Kumar, and Shikha Srivastava. Sehc: A benchmark setup to identify online hate speech in english. *IEEE Transactions on Computational Social Systems*, 10(2):760–770, 2022.

[47] Philip Gianfortoni, David Adamson, and Carolyn Rose. Modeling of stylistic variation in social media with stretchy patterns. In *Proceedings of the First Workshop on Algorithms and Resources for Modelling of Dialects and Language Varieties*, pages 49–59, 2011.

[48] Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. Chatgpt outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 120(30):e2305016120, 2023.

[49] Peter Glick and Susan T Fiske. Ambivalent sexism. In *Advances in experimental social psychology*, volume 33, pages 115–188. Elsevier, 2001.

[50] Umang Gupta, Jwala Dhamala, Varun Kumar, Apurv Verma, Yada Pruksachatkun, Satyapriya Krishna, Rahul Gupta, Kai-Wei Chang, Greg Ver Steeg, and Aram Galstyan. Mitigating gender bias in distilled language models via counterfactual role reversal. *arXiv preprint arXiv:2203.12574*, 2022.

[51] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. Don't stop pretraining: Adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*, 2020.

[52] Ari Aulia Hakim, Alva Erwin, Kho I Eng, Maulahikmah Galinium, and Wahyu Muliady. Automated document classification for news article in bahasa indonesia based on term frequency inverse document frequency (tf-idf) approach. In *2014 6th international conference on information technology and electrical engineering (ICITEE)*, pages 1–4. IEEE, 2014.

[53] Xiaochuang Han and Yulia Tsvetkov. Fortifying toxic speech detectors against veiled toxicity. *arXiv preprint arXiv:2010.03154*, 2020.

[54] Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. Toxigen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. *arXiv preprint arXiv:2203.09509*, 2022.

[55] Xingwei He, Zhenghao Lin, Yeyun Gong, Alex Jin, Hang Zhang, Chen Lin, Jian Jiao, Siu Ming Yiu, Nan Duan, Weizhu Chen, et al. Annollm: Making large language models to be better crowdsourced annotators. *arXiv preprint arXiv:2303.16854*, 2023.

[56] Xinlei He, Savvas Zannettou, Yun Shen, and Yang Zhang. You only prompt once: On the capabilities of prompt learning on large language models to tackle toxic content. *arXiv preprint arXiv:2308.05596*, 2023.

[57] Marlis Hellinger and Anne Pauwels. 21. language and sexism. *Handbook of language and communication: Diversity and change*, pages 651–684, 2008.

[58] Delia Irazu Hernandez Farias, Viviana Patti, and Paolo Rosso. Irony detection in twitter: The role of affective content. *ACM Transactions on Internet Technology*, 16(3), 2016.

[59] Fan Huang, Haewoon Kwak, and Jisun An. Is chatgpt better than human annotators? potential and limitations of chatgpt in explaining implicit hate speech. *arXiv preprint arXiv:2302.07736*, 2023.

[60] Karen Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21, 1972.

[61] Aditya Joshi, Pushpak Bhattacharyya, and Mark J Carman. Automatic sarcasm detection: A survey. *ACM Computing Surveys (CSUR)*, 50(5):1–22, 2017.

[62] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, volume 1, page 2, 2019.

[63] Hannah Rose Kirk, Wenjie Yin, Bertie Vidgen, and Paul Röttger. SemEval-2023 Task 10: Explainable Detection of Online Sexism. In *Proceedings of the 17th International Workshop on Semantic Evaluation*, Toronto, Canada, July 2023. Association for Computational Linguistics.

[64] Taja Kuzman, Igor Mozetič, and Nikola Ljubešić. Chatgpt: Beginning of an end of manual linguistic data annotation? use case of automatic genre identification, 2023.

[65] Jaida Langham and Kinnis Gosha. The classification of aggressive dialogue in social media platforms. In *Proceedings of the 2018 ACM SIGMIS Conference on Computers and People Research*, pages 60–63, 2018.

[66] Lingyao Li, Lizhou Fan, Shubham Atreja, and Libby Hemphill. " hot" chatgpt: The promise of chatgpt in detecting and discriminating hateful, offensive, and toxic comments on social media. *arXiv preprint arXiv:2304.10619*, 2023.

[67] Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*, 2023.

[68] Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A Smith, and Yejin Choi. Dexperts: Decoding-time controlled text generation with experts and anti-experts. *arXiv preprint arXiv:2105.03023*, 2021.

[69] Haixia Liu. Sentiment analysis of citations using word2vec. *arXiv preprint arXiv:1704.00177*, 2017.

[70] Ping Liu, Wen Li, and Liang Zou. Nuli at semeval-2019 task 6: Transfer learning for offensive language detection using bidirectional transformers. In *Proceedings of the 13th international workshop on semantic evaluation*, pages 87–91, 2019.

[71] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[72] Kate Manne. *Down girl: The logic of misogyny*. Oxford University Press, 2017.

[73] Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. Hatexplain: A benchmark dataset for explainable hate speech detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 14867–14875, 2021.

[74] Chris McCormick and Nick Ryan. Bert word embeddings tutorial. *URL: https://mccormickml. com/2019/05/14/BERT-word-embeddings-tutorial*, 2019.

[75] Shivangi Modi. Ahtdt-automatic hate text detection techniques in social media. In *2018 International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET)*, pages 1–3. IEEE, 2018.

[76] Saif M Mohammad, Parinaz Sobhani, and Svetlana Kiritchenko. Stance and sentiment in tweets. *ACM Transactions on Internet Technology (TOIT)*, 17(3):1–23, 2017.

[77] Ioannis Mollas, Zoe Chrysopoulou, Stamatis Karlos, and Grigorios Tsoumakas. Ethos: a multi-label hate speech detection dataset. *Complex & Intelligent Systems*, 8(6):4663–4678, 2022.

[78] Hamada A Nayel, Walaa Medhat, and Metwally Rashad. Benha@ idat: Improving irony detection in arabic tweets using ensemble approach. In *FIRE (Working Notes)*, pages 401–408, 2019.

[79] Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. Abusive language detection in online user content. In *Proceedings of the 25th international conference on world wide web*, pages 145–153, 2016.

[80] Oluwafemi Oriola and Eduan Kotzé. Evaluating machine learning techniques for detecting offensive and hate speech in south african tweets. *IEEE Access*, 8:21496–21509, 2020.

[81] Reynier Ortega-Bueno, Berta Chulvi, Francisco Rangel, Paolo Rosso, and Elisabetta Fersini. Profiling irony and stereotype spreaders on twitter (irostereo). 2021.

[82] Endang Wahyu Pamungkas, Valerio Basile, and Viviana Patti. Misogyny detection in twitter: a multilingual and cross-domain study. *Information Processing & Management*, 57(6):102360, 2020.

[83] Ji Ho Park and Pascale Fung. One-step and two-step classification for abusive language detection on twitter. *arXiv preprint arXiv:1706.01206*, 2017.

[84] Braja Gopal Patra, Kumar Gourav Das, and Dipankar Das. Multimodal author profiling for twitter. *Notebook for PAN at CLEF*, 2018.

[85] F Pedregosa et al. sklearn. feature_extraction. text. tfidfvectorizer. 2013.

[86] Haoruo Peng, Yangqiu Song, and Dan Roth. Event detection and co-reference with minimal supervision. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 392–402, 2016.

[87] Georgios K Pitsilis, Heri Ramampiaro, and Helge Langseth. Detecting offensive language in tweets using deep learning. *arXiv preprint arXiv:1801.04433*, 2018.

[88] Shahzad Qaiser and Ramsha Ali. Text mining: use of tf-idf to examine the relevance of words to documents. *International Journal of Computer Applications*, 181(1):25–29, 2018.

[89] Jing Qian, Mai ElSherief, Elizabeth M Belding, and William Yang Wang. Leveraging intra-user and inter-user representation learning for automated hate speech detection. *arXiv preprint arXiv:1804.03124*, 2018.

[90] Juliano Rabelo, Mi-Young Kim, Randy Goebel, Masaharu Yoshioka, Yoshinobu Kano, and Ken Satoh. Coliee 2020: methods for legal document retrieval and entailment. In *JSAI International Symposium on Artificial Intelligence*, pages 196–210. Springer, 2020.

[91] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[92] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[93] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[94] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

[95] Usha Nandini Raghavan, Réka Albert, and Soundar Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical review E*, 76(3):036106, 2007.

[96] Ashwin Rajadesingan and Huan Liu. Identifying users with opposing opinions in twitter debates. In *International conference on social computing, behavioral-cultural modeling, and prediction*, pages 153–160. Springer, 2014.

[97] Francisco Rangel, Anastasia Giachanou, Bilal Hisham Hasan Ghanem, and Paolo Rosso. Overview of the 8th author profiling task at pan 2020: Profiling fake news spreaders on twitter. In *CEUR Workshop Proceedings*, volume 2696, pages 1–18. Sun SITE Central Europe, 2020.

[98] Francisco Rangel, Paolo Rosso, Manuel Montes-y Gómez, Martin Potthast, and Benno Stein. Overview of the 6th author profiling task at pan 2018: multimodal gender identification in twitter. *Working Notes Papers of the CLEF*, pages 1–38, 2018.

[99] Francisco Rangel, Paolo Rosso, Martin Potthast, Benno Stein, and Walter Daelemans. Overview of the 3rd author profiling task at pan 2015. In *CLEF*, page 2015. sn, 2015.

[100] Amir H Razavi, Diana Inkpen, Sasha Uritsky, and Stan Matwin. Offensive language detection using multi-level classification. In *Advances in Artificial Intelligence: 23rd Canadian Conference on Artificial Intelligence, Canadian AI 2010, Ottawa, Canada, May 31–June 2, 2010. Proceedings 23*, pages 16–27. Springer, 2010.

[101] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

[102] Antonio Reyes and Paolo Rosso. Mining subjective knowledge from customer reviews: A specific case of irony detection. In *Proceedings of the 2nd workshop on computational approaches to subjectivity and sentiment analysis (WASSA 2.011)*, pages 118–124, 2011.

[103] Francisco Rodríguez-Sánchez, Jorge Carrillo-de Albornoz, and Laura Plaza. Automatic classification of sexism in social networks: An empirical study on twitter data. *IEEE Access*, 8:219563–219576, 2020.

[104] Francisco Rodríguez-Sánchez, Jorge Carrillo-de Albornoz, Laura Plaza, Adrián Mendieta-Aragón, Guillermo Marco-Remón, Maryna Makeienko, María Plaza, Julio Gonzalo, Damiano Spina, and Paolo Rosso. Overview of exist 2022: sexism identification in social networks. *Procesamiento del Lenguaje Natural*, 69:229–240, 2022.

[105] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[106] Timo Schick and Hinrich Schütze. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv preprint arXiv:2001.07676*, 2020.

[107] Timo Schick and Hinrich Schütze. It's not just size that matters: Small language models are also few-shot learners. *arXiv preprint arXiv:2009.07118*, 2020.

[108] Anna Schmidt and Michael Wiegand. A survey on hate speech detection using natural language processing. In *Proceedings of the fifth international workshop on natural language processing for social media*, pages 1–10, 2017.

[109] Tim Schopf, Daniel Braun, and Florian Matthes. Evaluating unsupervised text classification: zero-shot and similarity-based approaches. *arXiv preprint arXiv:2211.16285*, 2022.

[110] Roy Schwartz, Maarten Sap, Ioannis Konstas, Li Zilles, Yejin Choi, and Noah A Smith. The effect of different writing tasks on linguistic style: A case study of the roc story cloze task. *arXiv preprint arXiv:1702.01841*, 2017.

[111] Jaejin Seo, Sangwon Lee, Ling Liu, and Wonik Choi. Ta-sbert: Token attention sentence-bert for improving sentence representation. *IEEE Access*, 10:39119–39128, 2022.

[112] Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. The woman worked as a babysitter: On biases in language generation. *arXiv preprint arXiv:1909.01326*, 2019.

[113] Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. Towards controllable biases in language generation. *arXiv preprint arXiv:2005.00268*, 2020.

[114] Jiao Sun and Nanyun Peng. Men are elected, women are married: Events gender bias on wikipedia. *arXiv preprint arXiv:2106.01601*, 2021.

[115] Magdalena Szumilas. Explaining odds ratios. *Journal of the Canadian academy of child and adolescent psychiatry*, 19(3):227, 2010.

[116] Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. olmpics-on what language model pre-training captures. *Transactions of the Association for Computational Linguistics*, 8:743–758, 2020.

[117] Phoey Lee Teh, Chi-Bin Cheng, and Weng Mun Chee. Identifying and categorising profane words in hate speech. In *Proceedings of the 2nd International Conference on Compute and Data Analysis*, pages 65–69, 2018.

[118] Elise Fehn Unsvåg and Björn Gambäck. The effects of user features on twitter hate speech detection. In *Proceedings of the 2nd workshop on abusive language online (ALW2)*, pages 75–85, 2018.

[119] Francielle Vargas, Fabiana Rodrigues de Góes, Isabelle Carvalho, Fabrício Benevenuto, and Thiago Alexandre Salgueiro Pardo. Contextual-lexicon approach for abusive language detection. *arXiv preprint arXiv:2104.12265*, 2021.

[120] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[121] Voyage AI. Voyage-large-2 instructions. https://docs.voyageai.com/docs/embeddings, 2024. Accessed: 2024-06-15.

[122] Shuohang Wang, Yang Liu, Yichong Xu, Chenguang Zhu, and Michael Zeng. Want to reduce labeling cost? GPT-3 can help. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4195–4205, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[123] Zeerak Waseem. Are you a racist or am i seeing things? annotator influence on hate speech detection on twitter. In *Proceedings of the first workshop on NLP and computational social science*, pages 138–142, 2016.

[124] Zeerak Waseem and Dirk Hovy. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop*, pages 88–93, 2016.

[125] Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. Overview of the germeval 2018 shared task on the identification of offensive language. 2018.

[126] Ellery Wulczyn, Nithum Thain, and Lucas Dixon. Ex machina: Personal attacks seen at scale. In *Proceedings of the 26th international conference on world wide web*, pages 1391–1399, 2017.

[127] Guang Xiang, Bin Fan, Ling Wang, Jason Hong, and Carolyn Rose. Detecting offensive tweets via topical feature discovery over a large scale twitter corpus. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1980–1984, 2012.

[128] Peipeng Yu, Jiahan Chen, Xuan Feng, and Zhihua Xia. Cheat: A large-scale dataset for detecting chatgpt-written abstracts. *arXiv preprint arXiv:2304.12008*, 2023.

[129] Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. Predicting the type and target of offensive posts in social media. *arXiv preprint arXiv:1902.09666*, 2019.

[130] Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval). *arXiv preprint arXiv:1903.08983*, 2019.

[131] Dongwen Zhang, Hua Xu, Zengcai Su, and Yunfeng Xu. Chinese comments sentiment classification based on word2vec and svmperf. *Expert Systems with Applications*, 42(4):1857–1863, 2015.

[132] Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Gender bias in coreference resolution: Evaluation and debiasing methods. *arXiv preprint arXiv:1804.06876*, 2018.

[133] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16816–16825, 2022.

[134] Yiming Zhu, Peixian Zhang, Ehsan-Ul Haq, Pan Hui, and Gareth Tyson. Can chatgpt reproduce human-generated labels? a study of social computing tasks. *arXiv preprint arXiv:2304.10145*, 2023.

[135] Caleb Ziems, William Held, Omar Shaikh, Jiaao Chen, Zhehao Zhang, and Diyi Yang. Can large language models transform computational social science? *Computational Linguistics*, pages 1–55, 2024.

[136] Steven Zimmerman, Udo Kruschwitz, and Chris Fox. Improving hate speech detection with deep learning ensembles. In *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018)*, 2018.

Appendix A

Appendix

A.1 HateSpeechCorpus Details

Out of 3003 tweets, 2076 were annotated 'Hateful' and remaining 927 tweets were annotated 'Not Hateful' by the student annotators. The annotators achieved an average Cohen's Kappa score of 0.76. We paid each student $30/hour for doing the task.

A.2 Student Data Annotation Instructions

1. You will be working with the 'Sample.csv' file.

2. The first column contains the tweets to be annotated. The second column is 'Label'. There is also an additional column, 'Comments', for comments.

3. You will not be changing anything in the first column. These are the data downloaded from X (Twitter).

4. After reading each tweet from the first column, you will be deciding whether a tweet is 'Hateful' or 'Not Hateful', and you will be putting it in the second column (titled 'Label') for the corresponding tweet. So basically, the second column, 'Label', will contain two options: 1. Hateful 2. Not Hateful. You can use the following definition of hate [46] to decide whether a tweet is hateful or not:

   Hate: A hate tweet contains a directed insult(s), vulgar language to denigrate a target or words that instigate or support violence. Furthermore, the simple use of offensive

language such as slang and slurs on does not automatically result in a tweet of the type hate.

Non-Hate: All other tweets that do not fall in the hate category are non-hate tweets.

5. To decide whether the text is hateful, check specifically two points: 1. The target of the hate and 2. The keywords used in the text.

6. There could be some non-English words present in the tweets. In those cases, try to annotate by understanding the text's overall meaning.

7. The 'Sample Annotation.csv' file contains nearly 50 annotated tweets and the 'List of words.txt' file contains the list of words used to download the tweets. Please check these files before starting the annotation.

8. Lastly, you have an additional column, 'Comments' where you can add any comment if you want to. This section is totally optional.

A.3 Keywords for Downloading Tweets for HateSpeechCorpus

In this section, we enlist the keywords used for downloading tweets to construct HateSpeechCorpus. These keywords were chosen to capture a wide range of offensive and derogatory language, enabling us to compile a comprehensive dataset for studying hate speech patterns on social media. Table A.1 presents sample texts annotated by human annotators of various demographic groups and by two versions of a large language model (GPT-3.5 and GPT-4o). The table illustrates the variation and potential biases in annotations. Table A.2 presents a selection of sample texts from the dataset accompanied by human-provided annotations.

Keywords: bogan, bludger, hayseed/s, hoodrat/s, redneck/s, spiv/s, trailer trash, trailer park trash, yardie, yobbo/s, anchor baby, beaner/s, bull dyke/s, charva, chonkey, cripple/s, darky, eshays, gimp/s/y, goober, jigaboo, libtard, mongs, octaroon, peckerwood, quadroon, retard/s/ed, shemale, sperg, tard/s/ed, tranny, wetback, yokel, beaner, border hopper/s, border jumper/s, butt pirate/s, chinaman, ching chongs, coloreds, cunts, curry muncher, darkie/s, dindu nuffin, dune coon, dyke/s, fag, fagbag, fudgepacker, ghey/s, gypo, heebs, hilbilly, honkie/s, jiggaboo, jungle bunny, kikes, knacker, moon cricket/s, mud duck, mud shark, muzzie, ofay, papist/s, pickaninny, plastic paddy, pommie, tranny, whigger

Text	Dataset	Annotator	Annotation	Annotator	Annotation	LLM
Hello I'm calling from tech support *with Indian accent*	ETHOS	Asian	Hateful	Not Asian	Not Hateful	GPT-4o
having ADHD is so funny bc i think im a genius but i also know that im severely retarded	HateSpeechCorpus	Asian	Hateful	Not Asian	Not Hateful	GPT-4o
Killing me to see gays treated like that. I think lesbians are being more privileged by people	ETHOS	Asian	Not Hateful	Not Asian	Hateful	GPT-3.5
I thought kids with Down syndrome were not accepted by families!	ETHOS	Black	Hateful	Not Black	Not Hateful	GPT-4o
As a black person, I think that rappers should stop putting the n-word in the song. Plus, you should be proud of being White. I am proud of being black, so..	ETHOS	Black	Not Hateful	Not Black	Hateful	GPT-3.5

Continued on next page

Text	Dataset	Annotator	Annotation	Annotator	Annotation	LLM
All women love listening to me sperg out, they just hate everything else about me	HateSpeechCorpus	Black	Hateful	Not Black	Not Hateful	GPT-4o
Sweet and cute yet will cripple a beta with a snap of a finger.	HateSpeechCorpus	Black	Not Hateful	Not Black	Hateful	GPT-3.5
Fucking Canada...stupid. I think the cold shrinks their brains.	ETHOS	Female	Hateful	Not Female	Not Hateful	GPT-4o
Syria has a religion problem..	ETHOS	Muslim	Hateful	Not Muslim	Not Hateful	GPT-4o
I see the depression caused her to gain more weight.	ETHOS	Physical Disability	Hateful	No Disability	Not Hateful	GPT-4o
Is this guy actually autistic or Is he faking it	ETHOS	Mental Disability	Hateful	No Disability	Not Hateful	GPT-4o
What imaginary gender are you?	ETHOS	Muslim	Not Hateful	Not Muslim	Hateful	GPT-3.5
The Resistance avenge women and children killed by US and Allies !!!	ETHOS	Physical Disability	Not Hateful	No Disability	Hateful	GPT-3.5

Continued on next page

Text	Dataset	Annotator	Annotation	Annotator	Annotation	LLM
You look like someone who would do an electric wheelchair race with Stephen Hawking.	ETHOS	Physical Disability	Not Hateful	No Disability	Hateful	GPT-3.5
Eliminate their infrastructure and cripple them.	HateSpeechCorpus	Muslim	Hateful	Not Muslim	Not Hateful	GPT-4o
Like there was that one time where he was the gimp in the Armageddon and that other time, Magic Mike or whatever	HateSpeechCorpus	Physical Disability	Hateful	No Disability	Not Hateful	GPT-4o
Honestly the way football is turning is just sad, people are more bothered about filming themselves than the moments, so called celebs on the pitch during titles are won, and now gimps like this at a so called prestigious event. Embarrassing	HateSpeechCorpus	Mental Disability	Hateful	No Disability	Not Hateful	GPT-4o

Continued on next page

Text	Dataset	Annotator	Annotation	Annotator	Annotation	LLM
Houston is an important petro chemical hub. Bomb Houston and you could cripple multiple industries.	HateSpeechCorpus	Muslim	Not Hateful	No Muslim	Hateful	GPT-3.5
It's scary how "Technical Debt Cripples Companies And Threatens To Stifle Innovation" - learn more in this segment on with vFunction CEO	HateSpeechCorpus	Physical Disability	Hateful	No Disability	Not Hateful	GPT-3.5
You've wanted to visit for years gimp & shared your deepest needs online. It's time. Let's get started.	HateSpeechCorpus	Mental Disability	Not Hateful	No Disability	Hateful	GPT-3.5
Houston is an important petro chemical hub. Bomb Houston and you could cripple multiple industries.	HateSpeechCorpus	Muslim	Not Hateful	No Muslim	Hateful	GPT-3.5

Continued on next page

Text	Dataset	Annotator	Annotation	Annotator	Annotation	LLM
It's scary how "Technical Debt Cripples Companies And Threatens To Stifle Innovation" - learn more in this segment on with vFunction CEO	HateSpeechCorpus	Physical Disability	Hateful	No Disability	Not Hateful	GPT-3.5
You've wanted to visit for years gimp & shared your deepest needs online. It's time. Let's get started.	HateSpeechCorpus	Mental Disability	Not Hateful	No Disability	Hateful	GPT-3.5

Table A.1: Sample Texts with annotation biases while annotating by GPT-3.5 and GPT-4o.

Text	**Label (Annotation by Human)**
God help us. The UK is completely retarded. Labour/Conservative just 2 sides of the same coin.	Hateful
This has to be the most retarded fucking post I've seen all year on X	Hateful
My man had Keith Bogans at the shooting guard. Can't believe he was able to win with that	Not Hateful
The sport of kings, funded by punters robbed by spivs.	Not Hateful
She's trailer trash. That mouth is like fingernails on a chalkboard.	Hateful
Rhyhiem. I saw Yardie stick a gun in somebody mouth and that took me out of it.	Not Hateful
Ooh as a yobbo I'm a meat pie and sauce kind of guy	Not Hateful

Table A.2: Sample texts from HateSpeechCorpus with human annotation.