Cyberbullying Detection Model

Zeraiz Shabbir
Cloud Computing (CSC 4311)
Department of Computer Science
Atlanta, GA, United States

Abstract

Cyberbullying is a pervasive issue exacerbated by the widespread use of social media, affecting individuals across age groups with detrimental consequences. This project aimed to develop a machine learning model for detecting cyberbullying in tweets. Leveraging cloud-based infrastructure and machine learning techniques, including data preprocessing and model training, a model was constructed achieving 70.82% accuracy. Simple visualizations were generated to highlight common trends in cyberbullying tweets. The project contributes to combating cyberbullying by enabling prompt detection and intervention.

I. Introduction

Social media has become an integral part of modern communication, transcending age groups and geographical boundaries. Its pervasive reach and accessibility have transformed the way individuals interact and connect with one another. However, alongside the benefits of social media, there exists a darker side: cyberbullying. This insidious phenomenon can strike anyone, anywhere, and at any time, often cloaked in anonymity and enabled by the very platforms designed to facilitate communication.

The prevalence of cyberbullying is alarming, with significant impacts on individuals' mental health and well-being. According to recent statistics, 36.5% of middle and high school students report having experienced cyberbullying, while a staggering 87% have witnessed instances of cyberbullying. The effects of cyberbullying are far-reaching, ranging from decreased academic performance to profound psychological distress, including depression and even thoughts of self-harm (SOSNet).

In response to this pressing societal issue, the objectives of this project were twofold. Firstly, the goal was to develop a machine learning model capable of detecting instances of cyberbullying in social media content, with the objective of achieving an accuracy rate of above 50%. Secondly, the aim was to visualize trends in cyberbullying, shedding light on its prevalence and identifying common patterns and themes within cyberbullying discourse.

The structure of this report is organized into several sections to provide a comprehensive exploration of the research endeavor. Following this introduction, the motivation and background behind the project are detailed, emphasizing previous research while articulating how this study innovates and builds upon existing methodologies and insights. Subsequently, the design and implementation of the approach are outlined, highlighting the methodologies and tools employed in the endeavor. The evaluation section presents the results of the efforts, including the performance of the machine learning model and the insights gleaned from the trend analysis. Finally, a discussion of the findings engages in reflection on the implications of the research and avenues for future exploration in combatting cyberbullying.

Through this comprehensive examination, the aim is to contribute to the ongoing dialogue surrounding cyberbullying and foster a safer and more inclusive online environment for all users.

II. Motivation and Background

The motivation behind embarking on this project stems from both a personal desire for learning and a recognition of the profound societal importance of combating cyberbullying. As technology continues to evolve and permeate every aspect of our lives, it is imperative to equip oneself with the necessary tools and skills to navigate the digital landscape responsibly and ethically.

One notable inspiration for this project is the research conducted by Dalvi et al. (2020), as described in their paper presented at the 4th International Conference on Intelligent Computing and Control Systems. In their study, Dalvi et al. explored the development of a machine learning model to detect cyberbullying in social media content, specifically focusing on Twitter. They utilized machine learning algorithms such as Support Vector Machine (SVM) and Naive Bayes to classify tweets as either bullying or non-bullying, achieving promising results in terms of accuracy.

Building upon the foundation laid by Dalvi et al. and other researchers in the field, this project seeks to extend the existing body of knowledge by introducing novel methodologies and approaches to cyberbullying detection. While previous research primarily focused on textual analysis of tweets to identify instances of cyberbullying, this project takes a comprehensive approach by incorporating both textual analysis and contextual information.

One key differentiator of this project is the utilization of cloud infrastructure, specifically Apache Spark, for data preprocessing and analysis. By leveraging the distributed computing capabilities of Spark, the project aims to efficiently process large volumes of social media data and extract meaningful insights in real-time. Additionally, common trends across categories of cyberbullying have been visualized through the usage of matplotlib.

By addressing the limitations of previous approaches and introducing novel methodologies, this project aspires to contribute to the ongoing efforts to combat cyberbullying and create a safer online environment for all users.

III. Design and Implementation

This section outlines the design and implementation of the cyberbullying detection system, covering data acquisition, text preprocessing, model training, and evaluation. The development process was divided into five weeks, focusing on specific stages for each week:

Week 1: Data Acquisition

The first phase of the project involved acquiring a suitable dataset for cyberbullying detection. The dataset was sourced from Kaggle. It consisted of over 47,000 tweets labeled with six different categories related to cyberbullying. These categories were age, ethnicity, gender, religion, miscellaneous types of cyberbullying, and non-cyberbullying.

To ensure that the dataset was balanced and representative, approximately 8,000 samples were included for each category. This balancing strategy aimed to prevent any bias towards specific categories, ensuring that the model would learn from a diverse range of examples. The dataset was provided in CSV format and contained two columns. The first column, tweet_text, contained the tweet in its entirety, including any user mentions and other textual elements such as emojis. The second column, cyberbullying_type, contained the relevant category of cyberbullying with respect to the tweet_text in its row.

Week 2: Cloud Environment Setup and Text Preprocessing

During the second phase of development, efforts were directed towards setting up the cloud environment and implementing text preprocessing techniques. The cloud platform of choice was Databricks Community Edition, chosen for its ease of use and cost-free access. In exchange for its cost-free access, however, its utility is bounded by stringent memory constraints and fixed cluster configurations, precluding any adjustments, a limit which later posed challenges during development. A Python notebook with PySpark API was employed for preprocessing tasks, effectively allowing for the usage of Apache Spark in data analysis. The following preprocessing tasks were performed:

1. Lowercasing

```
tweets_df = tweets_df.withColumn("cleaned_tweet", lower(tweets_df["tweet_text"]))
```

This step converts all text to lowercase to ensure consistency in the dataset. Lowercasing prevents the model from treating words with different cases as different features. The lowercased data was saved to a new column, labeled cleaned_tweet.

2. Tokenization

```
tokenizer = Tokenizer(inputCol="cleaned_tweet", outputCol="tokens")
tweets_df = tokenizer.transform(tweets_df)
```

Tokenization breaks down the text into individual words or tokens. It splits the text into a list of tokens, enabling further processing at the word level. This step takes the cleaned_tweet column as input and transforms it into tokens then saves it to a new column labeled tokens.

3. Stop Words Removal

```
remover = StopWordsRemover(inputCol="tokens", outputCol="filtered_tokens")
tweets_df = remover.transform(tweets_df)
```

Stop words are common words that do not carry significant meaning for analysis. This step removes stop words from the tokenized text, reducing noise in the dataset and focusing on content-carrying words. The tokens column is used as input and is transformed into the filtered_tokens column.

4. Sentiment Analysis

```
def get_sentiment(tweet):
    analysis = TextBlob(tweet)
    return analysis.sentiment.polarity
```

```
udf_get_sentiment = udf(get_sentiment, FloatType())
tweets_df = tweets_df.withColumn("sentiment_score",
udf_get_sentiment("cleaned_tweet"))
```

Sentiment analysis assigns a sentiment score to each tweet, indicating the polarity of the text (positive, negative, or neutral). It is a float value ranging from -1 to 1. -1 indicates a very negative sentiment, 0 indicates a neutral statement, and 1 indicates a very positive sentiment. This score is achieved using the TextBlob Python library, which is a library for processing textual data and NLP tasks. It takes cleaned_tweet as input and creates the sentiment_score column based on that.

5. NGram Generation

```
ngram = NGram(n=3, inputCol="filtered_tokens", outputCol="trigrams")
tweets_df = ngram.transform(tweets_df)
```

NGram generation creates sequences of contiguous words of length 'n'. In this case, trigrams (sequences of three words) are generated from the filtered tokens. NGrams capture contextual information and enable the model to understand the relationships between words. In the evaluation phase of this project, unigrams and bigrams are used to visualize trends in data. For this step, the filtered_tokens column is taken as input, with trigrams being the output.

During this text preprocessing phase, each step was executed, sequentially enhancing the dataset with refined features essential for subsequent model training and evaluation stages. The resulting DataFrame encapsulates a comprehensive representation of the processed text features, sentiment scores, and NGrams, meticulously prepared for further analysis and insight generation.

Weeks 3 and 4: Model Training and Evaluation

With text preprocessing complete, the project entered its third phase – model training. These phases are combined since each stride in model iteration was coupled with a rigorous evaluation process to discern its efficacy. Initially, the aim was to train a model using a multinomial logistic regression algorithm, intended to categorize tweets across the spectrum of cyberbullying categories outlined in the dataset. However, the inherent constraints of Databricks Community Edition posed a formidable challenge, impeding the successful execution of this approach due to memory limitations. Despite resource constraints, a strategic decision was made to curtail the dataset size for each category, albeit at the risk of compromising model robustness. Nevertheless, this compromise was deemed necessary to glean insights into model performance and validate the effectiveness of the preprocessing pipeline. Yet, the resulting accuracy statistics fell short of expectations, necessitating a strategic pivot in approach.

Subsequently, the focus shifted towards constructing a binary classification model with the singular objective of detecting the presence of cyberbullying in tweets. Leveraging a logistic regression algorithm, the model training commenced anew, unencumbered by the need for data size restrictions. This strategic shift yielded more promising results, albeit with lingering opportunities for enhancement. However, the protracted training times, often exceeding an hour, presented a formidable hurdle, limiting the agility to fine-tune the model parameters.

Week 5: Model Refinement

Week 5 heralded a significant refinement phase necessitated by the constraints encountered with Databricks Community Edition. To circumvent the memory limitations inherent in the existing cloud environment, a strategic migration to an Azure virtual machine instance was orchestrated. By installing Python directly onto the virtual machine, the impediments posed by memory constraints were effectively surmounted. Furthermore, to augment the accuracy metrics, a transition to a Naïve Bayes algorithm was orchestrated, culminating in the realization of the most accurate model iteration to date. This strategic maneuver not only ameliorated the computational constraints but also underscored the iterative nature of the model development process, wherein adaptability and strategic pivots are paramount to success.

IV. Evaluation

The evaluation phase delved into the performance of each model iteration, shedding light on their efficacy in categorizing cyberbullying tweets. Here are the key performance metrics obtained:

1. **Multinomial Logistic Regression Algorithm Metrics:**
   - Accuracy: 0.321
   - F1 Score: 0.319

The multinomial logistic regression model demonstrated the lowest performance among the three models evaluated. Its accuracy and F1 score indicate that it struggled to effectively classify tweets into the correct categories, performing no better than random guessing. This could be attributed to the complexity of the dataset and the limitations of the algorithm in handling multi-class classification tasks.

2. **Binary Logistic Regression Algorithm Metrics:**
   - Area Under ROC: 0.616
   - Accuracy: 0.588
   - F1 Score: 0.588

The binary logistic regression model exhibited improved performance compared to the multinomial logistic regression model. With a higher accuracy and F1 score, it was better able to distinguish between cyberbullying and non-cyberbullying tweets. The binary classification task likely allowed the model to focus on a simpler distinction, leading to better performance.

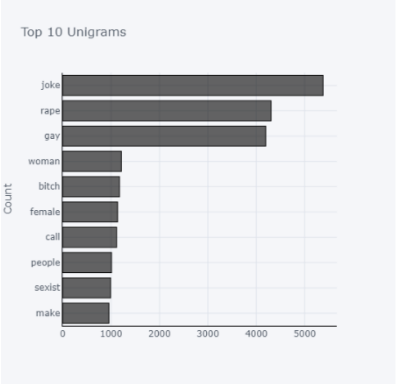
3. **Naïve Bayes Metrics:**
   - Area Under ROC: 0.721
   - Accuracy: 0.708
   - F1 Score: 0.707

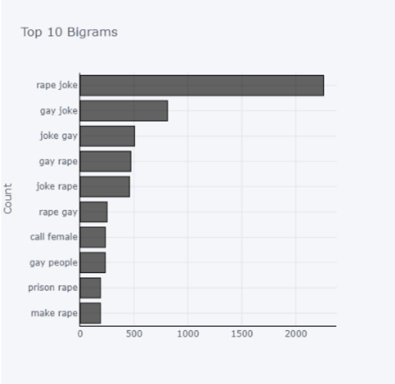
The Naïve Bayes model emerged as the top performer, surpassing both logistic regression models in accuracy and F1 score. Its superior performance could be attributed to its probabilistic approach and assumption of independence between features. Naïve Bayes models are well-suited for text classification tasks, making them particularly effective for identifying cyberbullying patterns in textual data.

Visualizations Analysis

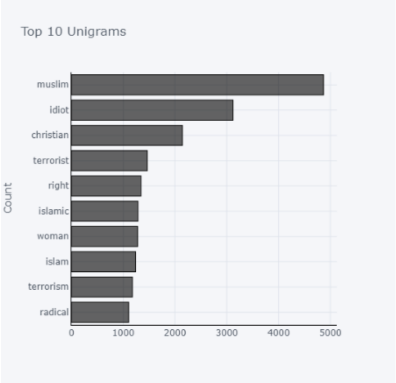
The visualizations in this project were created using the Matplotlib library in Python. After preprocessing the text data and categorizing tweets into different cyberbullying categories, the top 10 unigrams (single words) and top 10 bigrams (pairs of words) were extracted for each category, excluding the "not cyberbullying" category. These unigrams and bigrams were chosen to capture the most frequent and meaningful words and phrases associated with each category, providing insights into the language and themes prevalent in cyberbullying tweets. Matplotlib's versatile plotting functions were then utilized to generate bar charts displaying the frequency of these top unigrams and bigrams for each category.

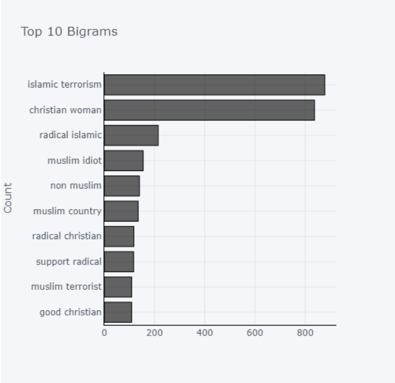
1. **Gender Visualizations**



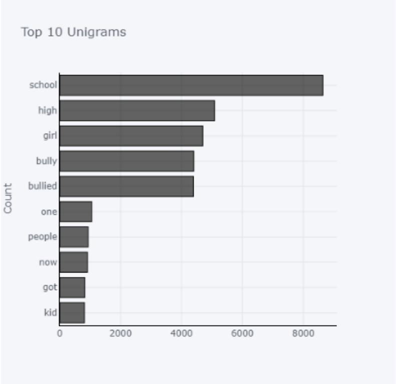


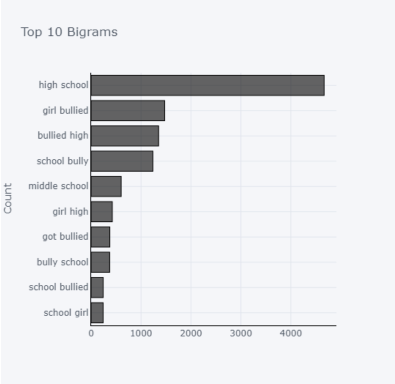
2. **Religion Visualizations**



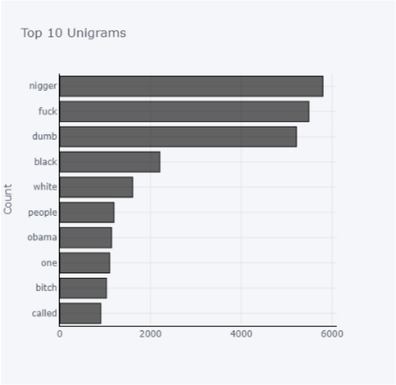


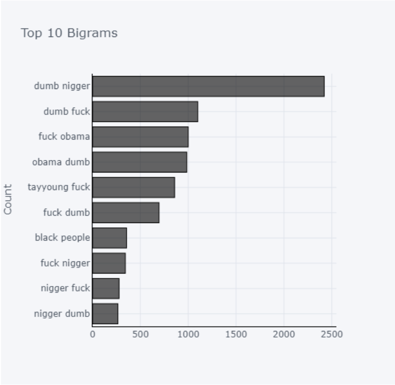
3. **Age Visualizations**



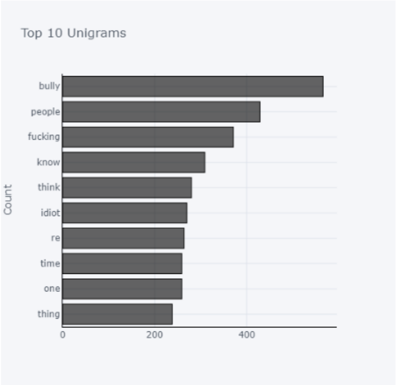


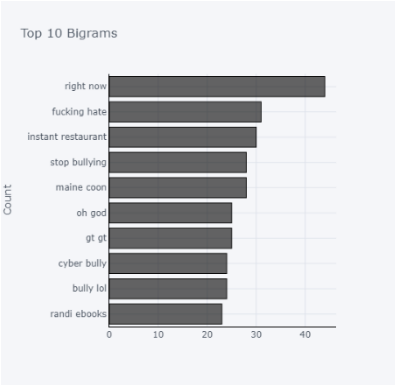
4. **Ethnicity Visualizations**





5. Other Visualizations





The visualizations unveiled distinct themes prevalent in cyberbullying tweets across various categories. Across gender-based cyberbullying, themes such as derogatory language and gender-specific insults emerged prominently, reflecting the pervasive use of offensive terms targeting individuals based on their gender identity. In religion-based cyberbullying, the visualizations highlighted the occurrence of religious slurs and references to extremist ideologies, indicating the propagation of hate speech and intolerance towards specific religious groups. Age-related cyberbullying themes centered around school-related bullying and instances of victimization, underscoring the vulnerability of young individuals to online harassment within educational settings. Ethnically driven cyberbullying depicted a pattern of racial epithets and racially charged language, indicating the prevalence of discriminatory attitudes and stereotypes. Moreover, the visualizations of cyberbullying categorized as "Other" revealed a diverse range of themes, including general harassment, profanity, and expressions of disdain, reflecting the multifaceted nature of online harassment beyond predefined categories.

V. Discussion

The discussion section encapsulates the culmination of findings, interpretations, and implications derived from the study's execution. This project embarked on a multifaceted exploration of cyberbullying detection, leveraging a combination of text analysis techniques and machine learning algorithms. The endeavor began with data acquisition from a Kaggle dataset, enriched with labeled cyberbullying tweets spanning various categories such as age, ethnicity, gender, religion, and others. Throughout the development stages, from text preprocessing to model training and evaluation, several notable insights surfaced.

The initial attempt at employing a multinomial logistic regression algorithm for multi-class classification yielded lackluster results, with accuracy barely surpassing random guessing. This outcome underscores the complexity of categorizing tweets across multiple cyberbullying dimensions and highlights the limitations of traditional algorithms in handling such nuanced tasks. The subsequent pivot to binary classification using logistic regression demonstrated improved performance, albeit with modest gains. However, it was the adoption of a Naïve Bayes algorithm that showcased the most promising results, achieving notable accuracy and F1 scores, along with a substantial increase in the area under the ROC curve.

A critical aspect of this project lies in the visualizations derived from the textual analysis of cyberbullying tweets. These visual representations elucidated prevalent themes and linguistic patterns across different categories of cyberbullying. From gender-specific derogatory language to racially charged epithets and religious slurs, the visualizations unveiled the pervasive nature of online harassment and the diverse forms it can take. Such insights are invaluable for understanding the underlying dynamics of cyberbullying and informing targeted interventions aimed at combating online harassment.

Despite the strides made in model performance and insights gained from visualizations, several challenges and limitations warrant acknowledgment. The project encountered memory constraints on the Databricks Community Edition platform, necessitating a shift to alternative cloud environments and impacting the scalability of model training. Moreover, the reliance on pre-existing labeled datasets introduces biases and may not fully capture the evolving nature of cyberbullying discourse in real-time settings. Future research endeavors could explore more sophisticated algorithms, ensemble techniques, or deep learning architectures to enhance model robustness and generalizability.

In conclusion, this project serves as a testament to the multifaceted nature of cyberbullying detection, highlighting the intersection of text analysis, machine learning, and societal implications. By leveraging advanced techniques and harnessing the power of data-driven insights, we take a step forward in understanding and addressing the pervasive issue of online harassment, ultimately striving towards a safer and more inclusive digital landscape.

References

J. Wang, K. Fu, C.T. Lu, "SOSNet: A Graph Convolutional Network Approach to Fine-Grained Cyberbullying Detection," Proceedings of the 2020 IEEE International Conference on Big Data (IEEE BigData 2020), December 10-13, 2020.

R. R. Dalvi, S. Baliram Chavan and A. Halbe, "Detecting A Twitter Cyberbullying Using Machine Learning," *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, 2020, pp. 297-301, doi: 10.1109/ICICCS48265.2020.9120893.