Northeastern University

Text Analysis of Tweets to Detect Cyberbullying Using NLP

Submitted By:

Bharat Ram, MS in Computer Science

Maheswar Raju, MS in Project Management

Kritika Gehlot, MPS Analytics

Responsible AI Hackathon

Table of Contents

1. Introduction

As social media usage continues to rise, so does the incidence of cyberbullying. Cyberbullying is a form of bullying that occurs through electronic technology, including social media platforms, and can cause significant emotional and psychological distress to victims. Traditional methods of identifying and addressing cyberbullying have proven to be insufficient, and there is a need for more effective strategies to detect and prevent this harmful behavior.

Extensive research in social sciences has investigated the reasons behind cyberbullying and its frequency, particularly among young adults and children. Studies based on evidence have revealed a correlation between cyberbullying experiences and suicidal thoughts in teenagers. Many studies provide insight into the magnitude of the issue and help raise awareness about it.

In this report, we explore the use of Natural Language Processing (NLP) techniques for detecting cyberbullying in tweets. NLP is a field of study that focuses on the interactions between natural language and computers, and it has become increasingly popular for analyzing text data. By applying NLP techniques to a dataset of tweets, we can extract relevant features and patterns that can be used to identify cyberbullying behavior.

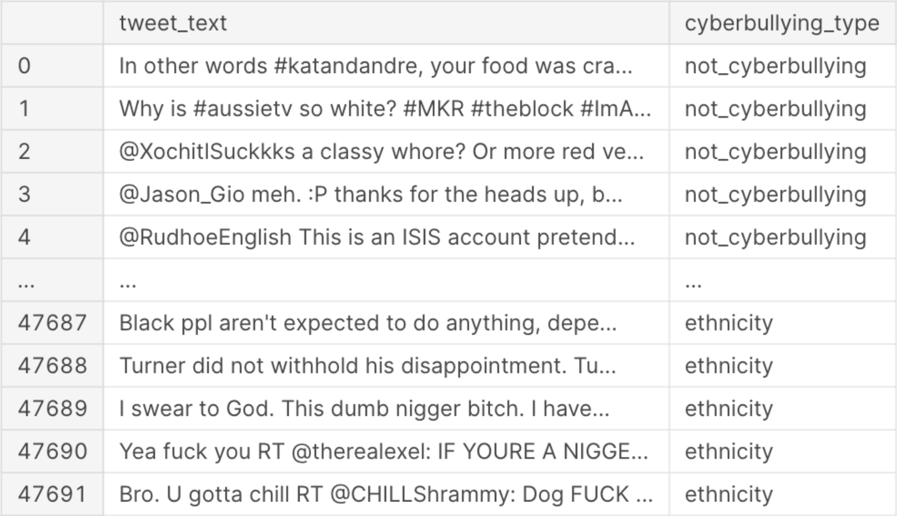
We will begin by discussing the challenges of cyberbullying detection. We will describe the data collection and preprocessing steps used in this study, followed by an explanation of the NLP techniques applied to the dataset. We will also present the results of our analysis, including the accuracy of our model in detecting cyberbullying behavior.

Overall, this report demonstrates the potential of NLP techniques for detecting cyberbullying behavior in social media. By leveraging these techniques, we can improve our ability to identify and address cyberbullying, ultimately leading to a safer and more positive online environment.

2. Dataset Description

The data set used in this study contains tweets that were collected using Twitter's API. The tweets were collected to include that contain certain keywords related to cyberbullying and some not. The data set contains approximately 47,692 tweets and categorized in categories: ‘not_cyberbullying’, ‘ethnicity’, ‘religion’, ‘gender’, ‘age’, and ‘other_cyberbullying’.

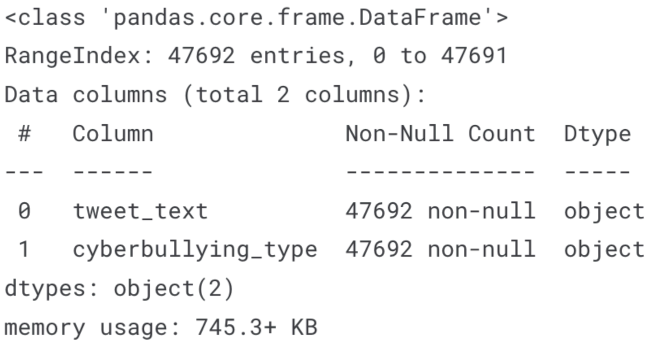
Here is the glimpse of our dataset below:



	tweet_text	cyberbullying_type
0	In other words #katandandre, your food was cra...	not_cyberbullying
1	Why is #aussietv so white? #MKR #theblock #ImA...	not_cyberbullying
2	@XochitlSuckkks a classy whore? Or more red ve...	not_cyberbullying
3	@Jason_Gio meh. :P thanks for the heads up, b...	not_cyberbullying
4	@RudhoeEnglish This is an ISIS account pretend...	not_cyberbullying
...	...	...
47687	Black ppl aren't expected to do anything, depe...	ethnicity
47688	Turner did not withhold his disappointment. Tu...	ethnicity
47689	I swear to God. This dumb nigger bitch. I have...	ethnicity
47690	Yea fuck you RT @therealexel: IF YOURE A NIGGE...	ethnicity
47691	Bro. U gotta chill RT @CHILLShrammy: Dog FUCK ...	ethnicity

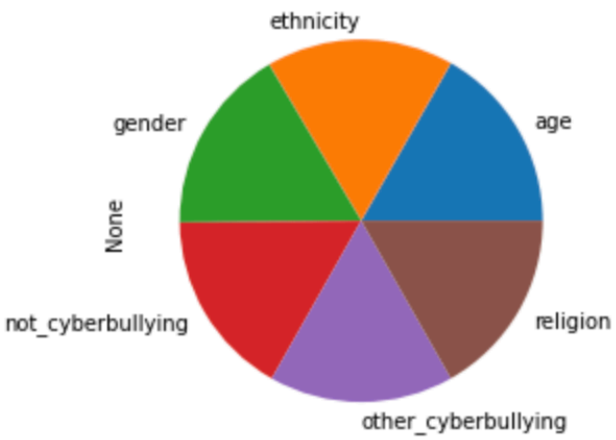
3. Exploratory Data Analysis (EDA)

In this section, we are going to perform some EDA to understand the nature of the data: EDA helps to identify the data types, the range of values, the distribution of data, and the relationship between variables. This helps to identify any inconsistencies in the data and provides a deeper understanding of the data. EDA helps to identify outliers and anomalies in the data, which can be important for detecting errors and understanding unusual behavior.



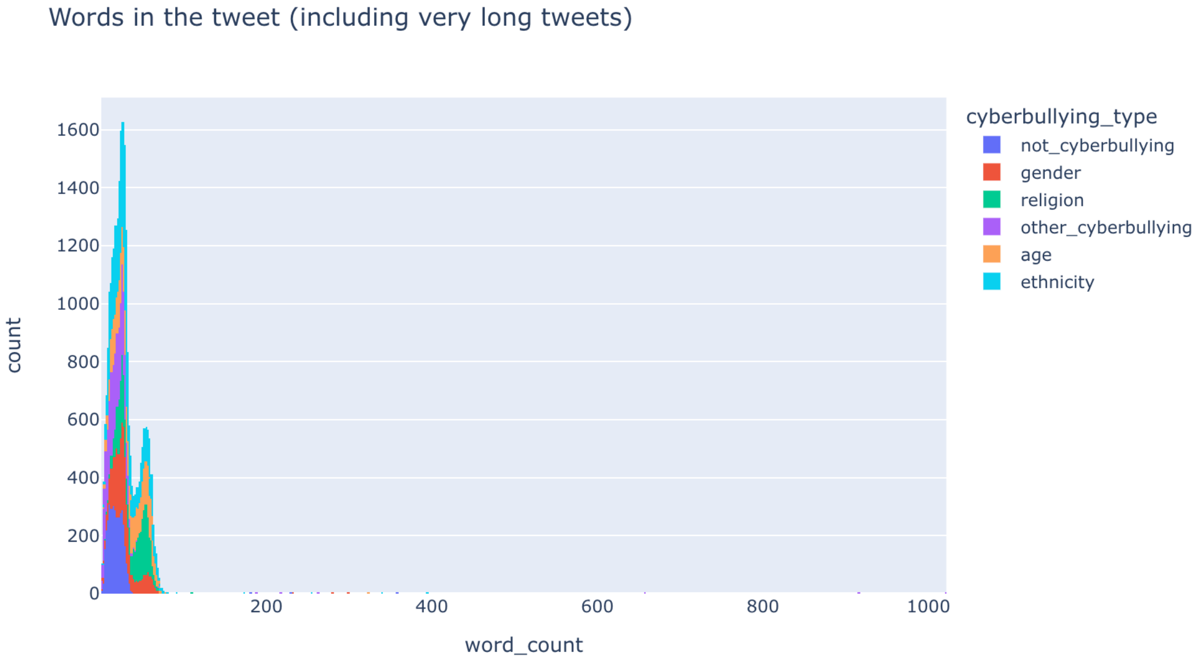
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 47692 entries, 0 to 47691
Data columns (total 2 columns):
 #   Column              Non-Null Count  Dtype 
---  ------              --------------  ----- 
 0   tweet_text          47692 non-null  object
 1   cyberbullying_type  47692 non-null  object
dtypes: object(2)
memory usage: 745.3+ KB
```

To get a better understanding of the data set, we conducted an exploratory data analysis. First, we looked at the distribution of the tweets by “cyberbullying_type”. We found that the dataset is quite balanced across all categories.

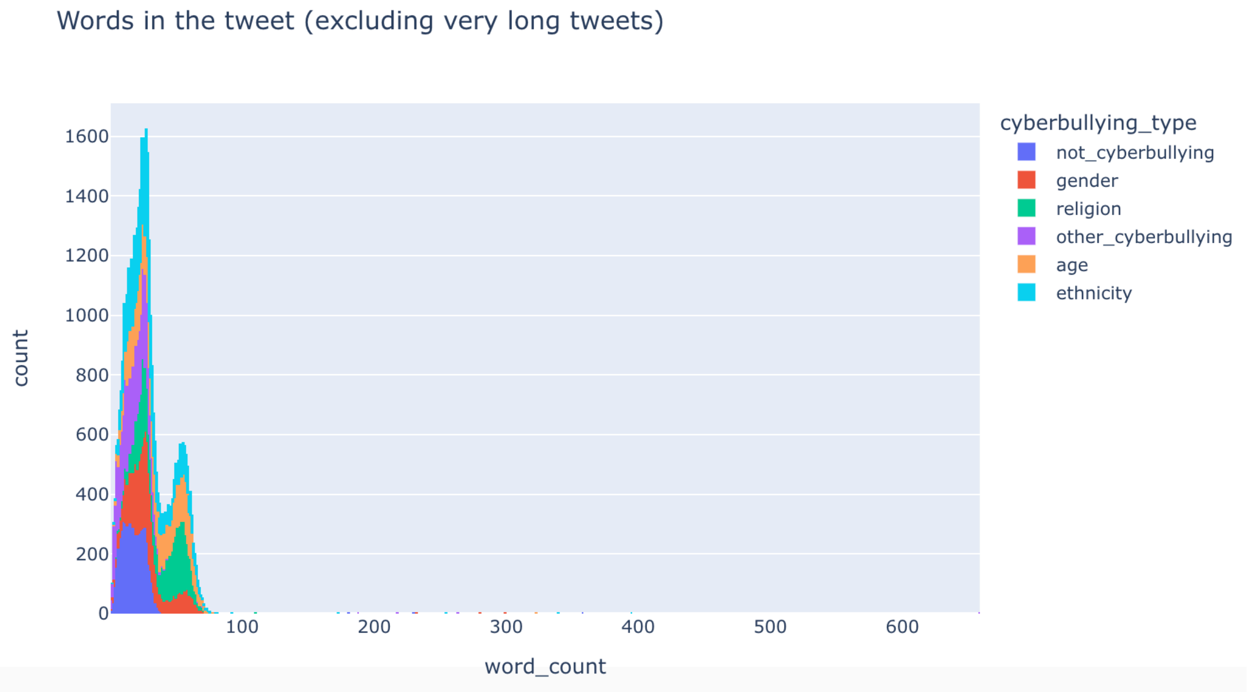


Next, we looked at the long are the messages in each category in the data set using stacked area chart.

We looked at words in tweets which included very long tweets by each category. We found that ‘ethnicity’ has highest count among all the cyber bullying categories.

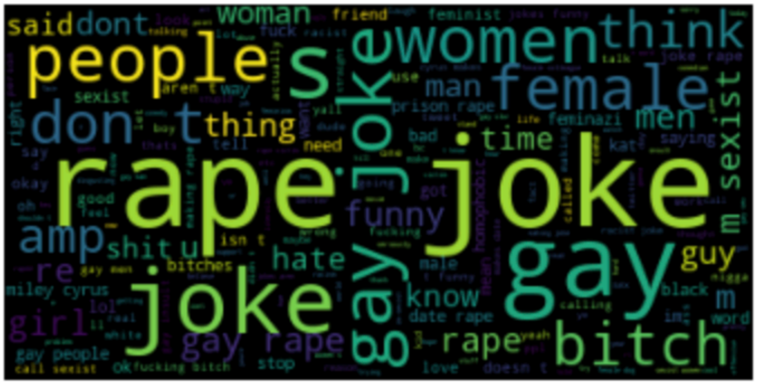


Then, we looked at words in tweets which excluding very long tweets by each category. We observed same thing that ‘ethnicity’ has highest count among all the cyber bullying categories.

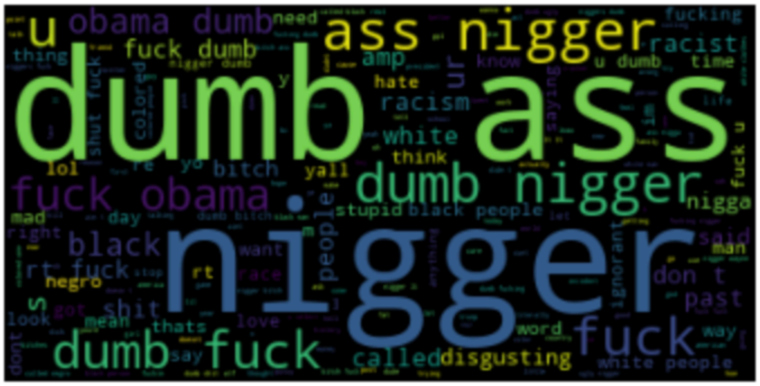


Then, we generated word cloud for every type of cyberbullying. We used it for text analysis as it can visually represents the most frequently occurring words in a text corpus. The size of each word in the cloud corresponds to its frequency or importance in the corpus.

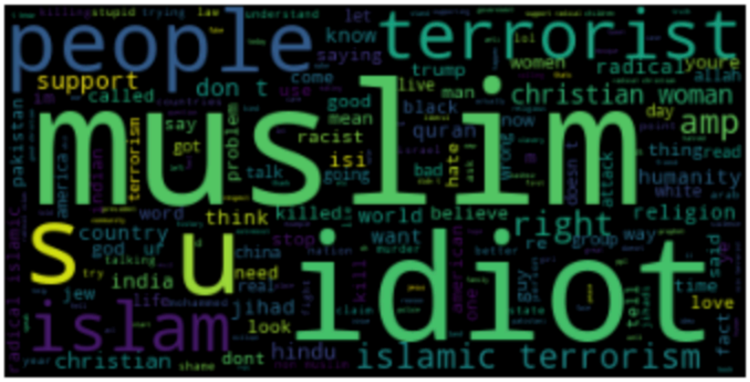
Gender



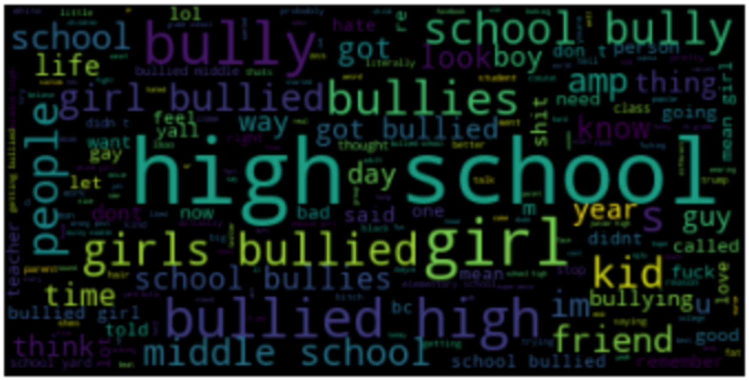
Ethnicity



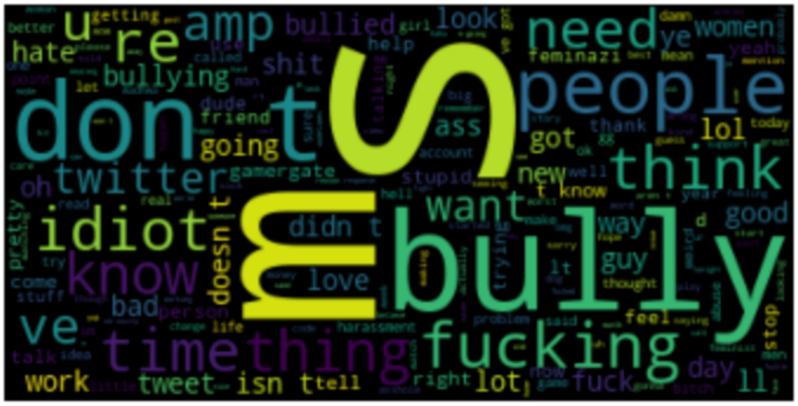
Religion



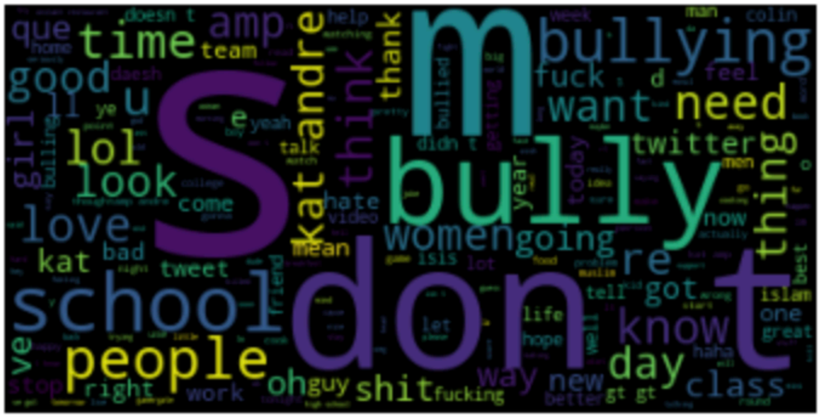
Age



Other



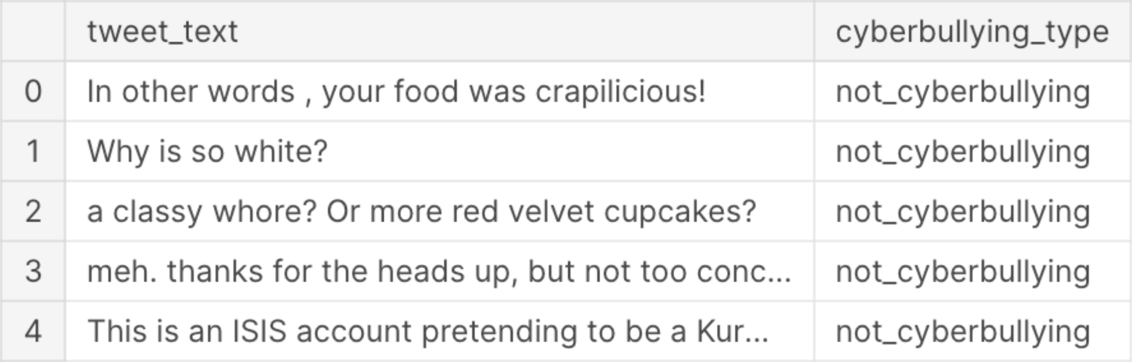
Non-bullying



4. Data Cleaning

4.1. Pre-Processing of Tweets

The Python library, Tweet-preprocessor, was created to facilitate the pre-processing of tweet data, which is a crucial step in developing Machine Learning systems that rely on such data. This library streamlines the cleaning, parsing, and tokenizing of tweets, making it easier for users to perform these tasks. Notably, Tweet-preprocessor supports various tweet elements, including URLs, hashtags, mentions, reserved words (such as RT and FAV), emojis, smileys, and both JSON and .txt file formats.



	tweet_text	cyberbullying_type
0	In other words , your food was crapilicious!	not_cyberbullying
1	Why is so white?	not_cyberbullying
2	a classy whore? Or more red velvet cupcakes?	not_cyberbullying
3	meh. thanks for the heads up, but not too conc...	not_cyberbullying
4	This is an ISIS account pretending to be a Kur...	not_cyberbullying

Here we can check it out, it removed all hashtags

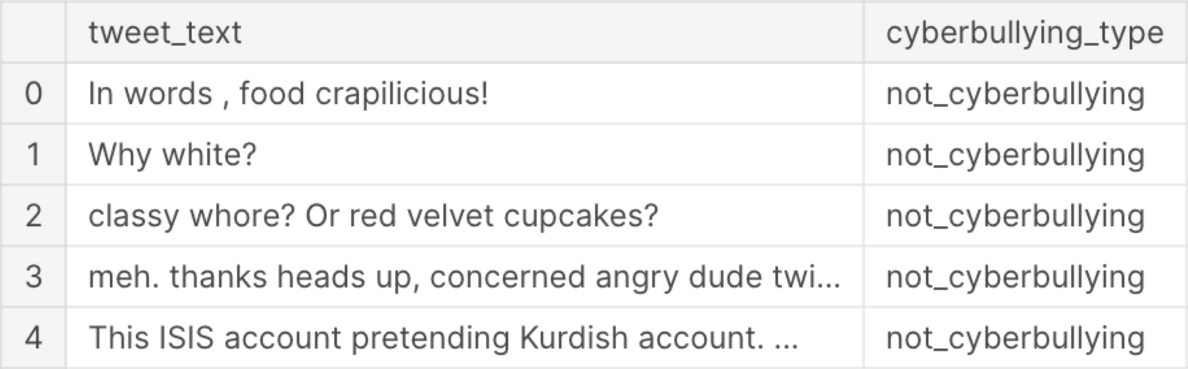
4.2. Removing Stop Word

It is essential to understand the concept of stop words in natural language processing. Stop words are a collection of frequently used words in any given language, including English. Removing stop words from text is essential in many applications, as it allows for better focus on the essential words that carry meaning in the text. By removing these frequently used words, we can potentially improve the accuracy and performance of machine learning models.

One of the significant benefits of removing stop words is that it can reduce the size of the dataset and decrease the time needed to train models. By eliminating unnecessary words, we can reduce the dimensionality of the data, making it easier to analyze and process.

Moreover, removing stop words can enhance the accuracy of classification models by eliminating irrelevant or redundant features. By focusing only on the important tokens, the model can better distinguish between different classes, leading to improved performance.
In summary, stop words are commonly used words in a language that can be removed from text to focus on meaningful words, reduce dataset size, and improve machine learning model performance.

Here is the data frame after removing stop words from tweets



	tweet_text	cyberbullying_type
0	In words , food crapilicious!	not_cyberbullying
1	Why white?	not_cyberbullying
2	classy whore? Or red velvet cupcakes?	not_cyberbullying
3	meh. thanks heads up, concerned angry dude twi...	not_cyberbullying
4	This ISIS account pretending Kurdish account. ...	not_cyberbullying

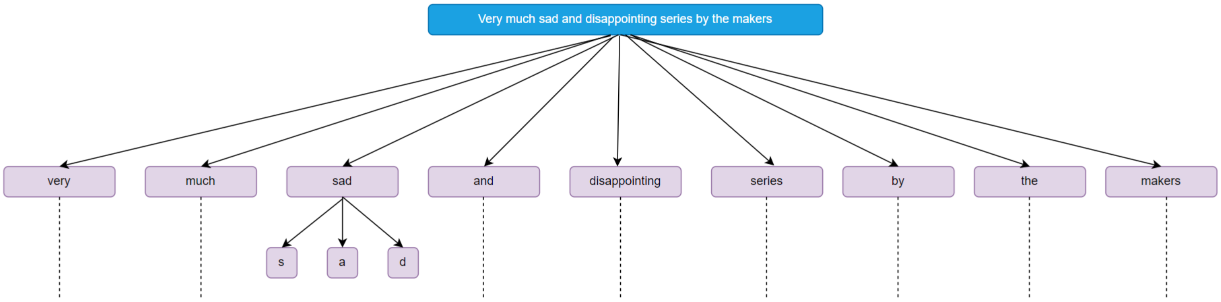
4.3. Tokenization

We have performed tokenization which is a technique used in natural language processing to break up a text into individual units of meaning, called tokens. In essence, tokenization is the process of splitting a sentence, paragraph, or document into smaller subunits such as words or phrases, which can then be analyzed further.

In most cases, tokenization involves splitting text based on whitespace or punctuation. For example, the sentence "The cat in the hat" could be tokenized into the following words: "The," "cat," "in," "the," and "hat."

Tokenization is a fundamental step in many natural language processing tasks, such as text classification, sentiment analysis, and machine translation. It helps to simplify the text by breaking it down into manageable units, which can then be analyzed or processed further.

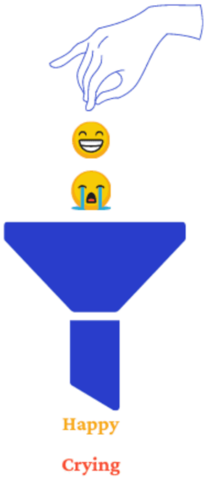
There are different levels of tokenization, ranging from word-level to character-level. Word-level tokenization is the most common and involves breaking text into individual words or phrases. Character-level tokenization, on the other hand, involves breaking text into individual characters or character n-grams.



We have used method word level tokenization to split a sentence into words. The output of word tokenization can be converted to Data Frame for better text understanding in machine learning applications.

4.4. Converting Emojis to Words

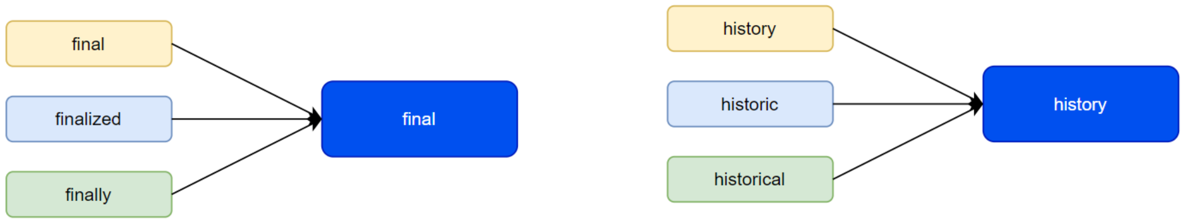
Converting emojis to words is a process of translating the visual representation of emojis into their corresponding textual description. Emojis are increasingly popular in online communication and are often used to convey emotions, sentiments, and reactions. However, their visual nature makes them difficult to process and analyze using traditional text analysis techniques. By converting emojis to words, it becomes possible to analyze and understand the meaning behind them. This process is typically done using pre-built libraries or classifiers that map the visual representation of emojis to their corresponding textual description. The resulting text can then be processed using natural language processing techniques such as sentiment analysis or topic modeling.



Converting Emojis to words

4.5. Lemmatization

Lemmatization is a technique used in natural language processing to reduce words to their base or dictionary form, known as a lemma. The goal of lemmatization is to normalize words so that different forms of the same word can be treated as one word, which helps improve the accuracy and efficiency of text analysis.



For example, in English, the word "run" can take on different forms such as "running," "ran," or "runs." By applying lemmatization, all these forms can be reduced to the base form "run."
Lemmatization works by using a dictionary or knowledge base of the language to identify the base form of a word, known as a lemma. The lemmatizer considers the part of speech of the word, as the lemma for a noun may be different from the lemma for a verb.
Compared to stemming, which reduces words to their root form by simply removing suffixes, lemmatization is a more sophisticated technique that produces better results by considering the context and meaning of words.

Overall, lemmatization is a valuable technique in natural language processing that helps to normalize words and improve the accuracy and efficiency of text analysis.

4.6. Label Encoding

Label Encoding is a process of converting categorical data into numerical format in Python. It is a common preprocessing technique used in machine learning to prepare data for training models. In this process, each unique category value in a categorical variable is assigned a unique numerical label. This transforms the data into a format that can be easily processed by machine learning algorithms.

In Python, label encoding can be performed using the LabelEncoder class from the sklearn.preprocessing module. The fit_transform() method of the LabelEncoder class can be used to transform the categorical data into numerical format.

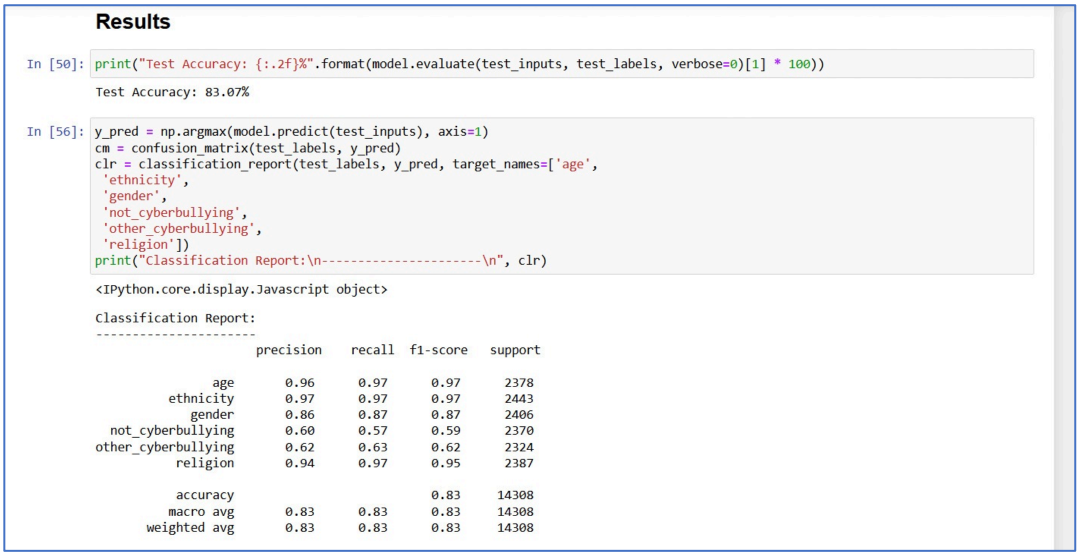
5. Training the Model and Results

We first build a text classification model to analyze Twitter tweets for cyberbullying. The model is built using the Keras API of TensorFlow.

The, we defined the neural network architecture using dense layers with rectified linear unit (ReLU) activation function and a final softmax activation function for classification into six categories of cyberbullying types. The model is trained using the sparse_categorical_crossentropy loss function, and accuracy is used as the evaluation metric.

After that, an embedding layer is added to the model architecture to encode the text data into numerical vectors, followed by a bidirectional GRU layer and a dense layer with a sigmoid activation function for binary classification.

The model is compiled and trained using the Adam optimizer and the EarlyStopping callback is used to prevent overfitting. The accuracy of the model on the test dataset is shown below, along with the classification report that includes precision, recall, and F1-score for each class.



Results

```
In [50]: print("Test Accuracy: {:.2f}%".format(model.evaluate(test_inputs, test_labels, verbose=0)[1] * 100))
```

```
Test Accuracy: 83.07%
```

```
In [56]: y_pred = np.argmax(model.predict(test_inputs), axis=1)
         cm = confusion_matrix(test_labels, y_pred)
         clr = classification_report(test_labels, y_pred, target_names=['age',
          'ethnicity',
          'gender',
          'not_cyberbullying',
          'other_cyberbullying',
          'religion'])
         print("Classification Report:\n----------------------\n", clr)
```

```
<IPython.core.display.Javascript object>

Classification Report:
----------------------
                      precision    recall  f1-score   support

                age       0.96      0.97      0.97      2378
          ethnicity       0.97      0.97      0.97      2443
             gender       0.86      0.87      0.87      2406
  not_cyberbullying       0.60      0.57      0.59      2370
other_cyberbullying       0.62      0.63      0.62      2324
           religion       0.94      0.97      0.95      2387

           accuracy                           0.83     14308
          macro avg       0.83      0.83      0.83     14308
       weighted avg       0.83      0.83      0.83     14308
```

Overall, this code snippet demonstrates a basic implementation of a text classification model using neural networks for analyzing Twitter tweets for cyberbullying.

6. Conclusions

The text analysis for Twitter tweets for cyber bullying has provided some interesting results. The classification report shows that the model was able to achieve an overall accuracy of 83%. The precision and recall scores for each class indicate that the model was able to perform well in identifying tweets related to age, ethnicity, religion, and not cyberbullying. However, the model's performance was relatively lower in identifying tweets related to gender and other cyberbullying.

Overall, the results show that text analysis can be an effective approach for identifying cyberbullying in Twitter data. However, further improvements can be made to the model to enhance its performance in identifying all types of cyberbullying with higher precision and recall scores. Nonetheless, the results of this study can be useful for organizations and researchers who are interested in identifying and preventing cyberbullying on social media platforms.

7. Challenges

While the results of the text analysis for Twitter tweets for cyberbullying are promising, there are still some challenges that need to be addressed. One of the main challenges in identifying cyberbullying on social media platforms is the use of slang and informal language that can be difficult to interpret. For instance, people who engage in cyberbullying may use a range of euphemisms, metaphors, or emojis to mask their abusive language, making it difficult for the algorithm to accurately identify such tweets.

Another challenge is related to the contextual nature of cyberbullying. In some cases, tweets may appear harmless on their own, but can be part of a broader pattern of cyberbullying when considered in the context of other tweets or online interactions. Additionally, people who engage in cyberbullying may use subtle or implicit forms of aggression, such as exclusion, rumor-spreading, or manipulation, which can be difficult to detect using standard text analysis approaches.

Despite these challenges, text analysis remains an important tool for identifying cyberbullying on social media platforms. The results of the classification report indicate that the model was able to achieve high precision and recall scores for some types of cyberbullying, such as age, ethnicity, religion, and not cyberbullying. These results suggest that text analysis can be an effective approach for identifying some types of cyberbullying, which can be used by organizations and researchers to develop targeted interventions to prevent or mitigate cyberbullying.