**Enhancing Social Media Sentiment Analysis with Emoji and Emoticon Embeddings for Mitigating Online Toxicity**

Pranshu Choubey  
School of Computer Science and Engineering  
Vellore Institute of TechnologyChennai, India  
pranshu.choubey2020@vit.ac.in

Sandhya. M   
School of Computer Science and Engineering   
Vellore Institute of TechnologyChennai, India  
sandhya.m@vit.ac.in

*Abstract*—**Social media platforms have become vital channels for individuals to express themselves and engage in communication on a global scale. However, along with the benefits of connectivity, they also harbor significant challenges, particularly concerning the proliferation of online toxicity and hate speech. The prevalence of such harmful content underscores the urgent need for effective moderation strategies to cultivate a safer and more inclusive digital environment. Traditional sentiment analysis methods, while useful in identifying positive and negative sentiments in text data, often struggle to accurately capture the nuances of online communication. One significant limitation lies in their inability to adequately account for the contextual information conveyed by emojis and emoticons. These visual elements play a crucial role in augmenting textual content with emotional cues, adding layers of meaning that are essential for understanding the true sentiment behind a message. The proposed research aims to address this gap by developing an innovative methodology that integrates emoji and emoticon embeddings into hate speech detection models. By leveraging these visual cues alongside textual data, the project seeks to enhance the accuracy and effectiveness of sentiment analysis in identifying and flagging toxic content on social media platforms. Central to the study is the development of a function designed to process tweets containing emojis, extracting and generating embeddings that preserve the contextual information conveyed by these visual elements. These embeddings serve as a bridge between the textual and visual components of social media messages, enabling hate speech detection models to analyze both aspects simultaneously.**

***Keywords—Action Recognition, Pose Detection, Feedback, Football, GenAI***

# Introduction

In the contemporary digital landscape, characterized by prolific social media engagement, platforms like Twitter, Instagram and Facebook serve as virtual arenas where individuals express their opinions, share information, and engage in discourse on a global scale. However, amidst the vast sea of online communication, there exists a pressing concern: the proliferation of toxic content, hate speech, and negative sentiments that permeate these platforms, particularly evident in the brevity and spontaneity of tweets. The ability to discern and mitigate such harmful expressions has become a paramount challenge for social media platforms, users, and researchers alike.

This project, titled "Enhancing Social Media Sentiment Analysis through Emoji Embeddings," delves into the intricate realm of sentiment analysis on Twitter, with a specific focus on integrating emojis and emoticons to improve the detection of hate speech and toxic content within tweets. Emojis and emoticons, once considered mere embellishments in digital conversations, have evolved into powerful tools for conveying emotions, attitudes, and sentiments within the constrained format of tweets. However, traditional sentiment analysis methods often overlook the rich contextual information encoded within these visual elements, leading to limited accuracy in detecting toxicity and negative sentiment. The primary objective of this project is to bridge this gap by proposing an innovative methodology that harnesses the potential of emoji and emoticon embeddings to enhance sentiment analysis models tailored for Twitter data. By integrating these visual cues with textual data extracted from tweets, the project aims to develop a more nuanced understanding of online discourse, particularly in identifying and flagging hate speech and toxic content within the short and dynamic nature of tweets.

The significance of this endeavor lies in its potential to contribute to the creation of safer and more inclusive online spaces within the Twitter ecosystem. As Twitter continues to play a pivotal role in shaping public discourse and influencing societal attitudes, addressing the issue of online toxicity is imperative for fostering healthy digital environments. By leveraging advanced sentiment analysis techniques augmented with emoji embeddings, this project seeks to empower platform moderators, policymakers, and users with the tools necessary to combat hate speech and promote constructive dialogue within the unique constraints of Twitter's platform.

The journey towards achieving this goal encompasses several key components. First and foremost is the collection of diverse Twitter data, comprising tweets that span a wide range of topics, sentiments, and user demographics. This dataset serves as the foundation upon which the subsequent phases of the project are built. Next comes the design and implementation of a novel model architecture capable of integrating emoji embeddings into existing sentiment analysis frameworks tailored specifically for tweets. This phase involves leveraging machine learning and natural language processing techniques to develop a robust and scalable solution optimized for Twitter data.

Throughout the project lifecycle, emphasis is placed on rigorous analysis and evaluation to assess the effectiveness and performance of the proposed methodology within the context of Twitter. By systematically measuring key metrics such as accuracy, precision, recall, and F1 score, the project aims to provide empirical evidence of its efficacy in detecting hate speech and toxic content within tweets. Additionally, qualitative analysis of the results offers insights into the contextual nuances of online communication within the Twitter platform, shedding light on the complex interplay between language, emotion, and social dynamics.

The integration of emoji embeddings represents a promising avenue for enhancing social media sentiment analysis on Twitter and combating online toxicity within its unique format and constraints. By combining computational methods with insights from linguistics, psychology, and sociology, this project endeavors to contribute to the ongoing efforts to create a more inclusive and safer digital spaces within the Twitter ecosystem.

# Background study

## Objective

The primary objective of this research project is to advance the field of sentiment analysis within the context of social media by leveraging emojis and emoticons embeddings to specifically address and mitigate online toxicity. The following research objectives outline the key goals and milestones for achieving success in this endeavor.

Design and implement a novel methodology for enhancing sentiment analysis on social media platforms by integrating emoji and emoticon embeddings into hate speech detection models. Create a function capable of processing tweets containing emojis, generating embeddings to convert text+emoji tweets into text-only while preserving emoji context. This involves developing techniques to disambiguate visual elements when conveying hate, dark sentiments, or other forms of online toxicity. Utilize the generated embeddings to enhance existing sentiment analysis models, thereby improving their accuracy in detecting hate speech and toxic content. Enhancements include refining algorithms for real-time processing, scalability, and integrating contextual sensitivity for better interpretation of hate speech within the broader context of social media conversations. Improve existing sentiment analysis models to better identify and flag instances of online toxicity, hate speech, and dark sentiments by incorporating both textual and visual elements. This involves exploring multimodal fusion strategies to seamlessly integrate textual and visual features for a more comprehensive understanding of toxic expressions.Conduct thorough evaluation and analysis of the developed methodology to assess its effectiveness in identifying and flagging hate or dark comments/tweets on social media.

## Literature Review

In this literature review, we have explored a diverse range of research papers focused on enhancing sentiment analysis and addressing online toxicity through the incorporation of emojis and emoticons. These studies underscore the significance of emojis and emoticons as powerful indicators of sentiment and emotion in social media texts, particularly in platforms like Twitter where communication is often concise and informal.

Liu et al. [1] investigated the utilization of emojis in sentiment analysis of online Chinese texts, with the objective of enhancing accuracy and comprehending sentiment changes amidst the COVID-19 pandemic. The research presents CEmo-LSTM, an enhanced emoji-embedding model utilizing Bi-LSTM, and assesses its efficacy through various sentiment analysis techniques, encompassing rule-based and classification algorithms. The study contrasts the performance of CEmo-LSTM with prevalent sentiment analysis models such as LSTM and BERT across diverse experimental setups. Emphasizing the influence of emojis on sentiment analysis algorithms, the research examines different emoji usages' effectiveness in texts. Additionally, it evaluates the pandemic's impact on individual moods employing the CEmo-LSTM algorithm. The study reveals emojis' efficacy in augmenting sentiment analysis algorithm accuracy and underscores the pandemic's adverse effect on individual sentiments, culminating in a surge of passive emotions. Proposing a pioneering emoji-embedding algorithm grounded in emoji usage for sentiment analysis, the research advocates further investigation into emoji usage patterns and contexts to refine the CEmo-LSTM algorithm's performance.

Surikov et al. [2] proposes employing a word-vector and emoji/emoticon embedding model for sentiment analysis on concise informal texts sourced from social media platforms. The research utilizes a dataset comprising emotional indicators (emojis and emoticons) with associated textual meaning labels like "happiness" or "sadness". A word-vector model is constructed using a Twitter corpus to generate embeddings for both words and emojis/emoticons. These embeddings are unified within a single semantic space, and a logistic regression classifier is trained on this combined embedding model to forecast sentiment labels. The model achieves an average accuracy of 71.5% for two sentiment categories (positive and negative), surpassing baselines that lack emoji/emoticon embeddings. Performance assessment employs metrics including accuracy, F1 score, and area under the ROC curve (AUC) on a Twitter test dataset. The study concludes that the ensemble approach of integrating word and emoji/emoticon embeddings proves effective for sentiment analysis on succinct informal texts from social media.

Ayvaz et al. [3] examined the utilization of emoji characters on Twitter and their impact on sentiment analysis and text mining. Twitter serves as the data source due to its capability for real-time access to public opinions. Data is gathered for two events - New Year's Eve (positive) and the Istanbul attack (negative) - and subjected to analysis both with and without considering emojis. An R-based sentiment analysis application is developed to evaluate tweets using the SentiWordNet sentiment dictionary, which comprises positive and negative English words. Emojis are incorporated into the dictionaries based on their sentiment scores. Inclusion of emojis results in a 2% increase in positive tweet scores for the positive event. Conversely, for the negative event, emojis lead to a decrease of 1.4% in neutral tweets and a slight increase in positive tweets. The study concludes that incorporating emojis in sentiment analysis yields higher sentiment scores compared to disregarding emojis, showcasing how social media data coupled with emojis can enrich sentiment analysis endeavors.

Vashisht et al. [4] conducted a comprehensive review of trends in emoticon (emoji)-based sentiment analysis research spanning from 2005 to 2019. Various studies utilizing emoticons for sentiment classification across tweets, blogs, and other textual sources were examined. Common machine learning algorithms such as Naive Bayes, maximum entropy classifiers, SVMs, k-NN, and SGD were employed. Features encompassed emoticons, word unigrams and bigrams, part-of-speech tags, and sentiment lexicons. Metrics for performance evaluation included accuracy, precision, recall, and F1 score. Some studies reported achieving accuracy rates of up to 92-94% for sentiment classification utilizing emoticons. Identified challenges involved the absence of a standardized emoticon lexicon, ambiguity in emoticon interpretations, scarcity of emoticons in concise texts, and the inadequacy of n-grams in effectively modeling emojis. The study concluded that integrating emoticons with existing machine learning and lexicon-based approaches can enhance sentiment analysis accuracy compared to employing those techniques independently. Nonetheless, further research is warranted to tackle the identified challenges.

Guibon et al. [5] delved into the definitions of emojis and emoticons, categorizing various usages of emojis within context, such as sentiment enhancement and modification. The paper explored applying different natural language processing and machine learning techniques for sentiment analysis on texts containing emojis, including n-gram models and SVM classifiers. A proposed methodology utilized an existing sentiment lexicon (ESR) to initially detect the sentiment of emojis. Subsequently, tools like SentiStrength were employed to detect the sentiment of sentences, followed by a comparison to identify the type of emoji usage. Models were developed to automatically discern emoji usage types on annotated corpora, aiming to enhance sentiment analysis by considering different impacts of emojis based on their usage. Potential applications discussed encompassed utilizing identified emoji meanings to empower emotional conversational agents and avatars. The paper provided relevant background on existing work analyzing emoticon usage for sentiment analysis, suggesting potential extensions to emojis.

Wolny et al. [6] explored the utilization of emoticons and emoji ideograms for sentiment analysis on Twitter data, acknowledging the challenges posed by the informal language prevalent on social media platforms. Existing sentiment analysis methods primarily focus on polarity (like/dislike), yet human emotions exhibit greater complexity. Emoticons and emojis serve to capture this nuanced emotional spectrum. Twitter data was collected using Twitter APIs and pre-processed to filter tweets based on relevant hashtags and user mentions. The Python library Tweepy facilitated the connection to the Twitter Streaming API for downloading tweets in JSON format, subsequently converted to CSV or database formats. Different emoticons and emoji symbols were categorized into emotion classes such as happiness, sadness, and anger based on existing literature on basic emotions. The frequency of various emoticons/emoji in tweets was analyzed to discern sentiment and conduct multi-class emotion classification using machine learning algorithms. Incorporating emoticons and emoji analysis alongside natural language processing significantly enhanced precision in identifying various emotions expressed in tweets.

Jagadishwari et al. [7] aimed to detect sentiments expressed in social media posts employing machine learning models. The utilized dataset comprised 300,000 tweets with sentiment scores and text, split into 70% for training and 30% for testing. Data preprocessing steps encompassed removing punctuation, HTML tags, stopwords, lemmatization, and lowercasing. Four machine learning algorithms: Bernoulli Naive Bayes, Multinomial Naive Bayes, Linear Regression, and SVM were implemented. Initially, models were trained solely with text, followed by training with both text and emoticons to assess the emoticons' impact. Performance evaluation utilized classification reports, measuring accuracy, precision, recall, and F1 score. The Naive Bayes models, particularly Multinomial Naive Bayes, exhibited the best performance, achieving up to 89% accuracy. Emoticons were observed to have a negligible effect on model performance. The paper concluded that the Naive Bayes family of classifiers, particularly Multinomial Naive Bayes, are well-suited for sentiment analysis tasks based on their performance in this study.

Bhaskaran et al. [8] propose a methodology for detecting toxic online conversations utilizing a multi-task learning approach with convolutional neural networks (CNNs). The dataset comprises Wikipedia comments annotated for toxicity. Pre-processing involves text normalization, tokenization, and padding. The model architecture comprises an embedding layer, CNN layer, max-pooling layer, and softmax output layer. Training is conducted with categorical cross-entropy loss. In addition to toxicity detection, the model predicts comment length and identifies personal attacks as auxiliary tasks within a multi-task learning setup. The model reached an F1 score of 0.89 for toxicity detection, outperforming single-task baselines. Including auxiliary prediction tasks enhances the main toxicity detection task. The methodology demonstrates the effectiveness of multi-task learning with CNNs for online toxicity detection. Additionally, the incorporation of auxiliary tasks aids in capturing more context about problematic conversations.

From examining various methodologies and techniques proposed in the reviewed papers, it is evident that emojis and emoticons play a crucial role in augmenting sentiment analysis tasks. Embracing these visual elements enables sentiment analysis models to capture nuanced emotional cues that may not be adequately conveyed through text alone. Moreover, leveraging emojis and emoticons can significantly improve the performance of sentiment analysis models, especially in contexts like software engineering and social media, where traditional text-based methods may fall short due to domain-specific jargon or data scarcity.

Drawing insights from the reviewed literature, this project aims to address the limitations of existing sentiment analysis models by incorporating emojis and emoticons. By developing a novel function that processes tweets with emojis and generates embeddings to convert text+emoji tweets into text-only while preserving emoji context, the project endeavors to enhance hate speech detection and sentiment analysis on social media platforms like Twitter.

By integrating emojis and emoticons into sentiment analysis models, the project seeks to not only improve the accuracy of hate speech detection but also enhance existing sentiment analysis models' ability to discern subtle emotional nuances in social media texts. This innovative approach holds promise for mitigating online toxicity and fostering a safer and more inclusive online environment.

Overall, the findings from this literature review provide valuable insights and serve as a solid foundation for this project, offering a roadmap for leveraging emojis and emoticons to enhance sentiment analysis and combat online toxicity effectively. Through the integration of visual cues into sentiment analysis models, the project endeavors to contribute to the advancement of affective computing and promote positive online interactions.

# Methodology

This section outlines the systematic approach employed to develop and evaluate a hate speech detection system using machine learning techniques. Detecting hate speech in online content is a complex task that requires a multi-faceted approach encompassing data preprocessing, exploratory data analysis, feature engineering, model building, evaluation, and optimization. It provides a detailed overview of each step undertaken in the development of the hate speech detection system.

**3.1 DATA COLLECTION AND PRE-PROCESSING**

The first step in the methodology is the collection of data, which involves obtaining a dataset that contains tweets along with their corresponding labels indicating whether they are hate speech or not. The dataset used in this project is acquired from an external source and consists of three columns: **id**, **label**, and **tweet**.

After acquiring the dataset, the next step is data preprocessing. This involves several steps to clean and prepare the text data for analysis and model training. The preprocessing steps include:

* **Lowercasing**: Convert all text to lowercase to ensure consistency in text analysis.
* **Removal of URLs**: Use regular expressions to remove URLs from the tweets as they do not contribute to the sentiment analysis.
* **Removal of Mentions and Hashtags**: Use regular expressions to remove Twitter mentions (@username) and hashtags (#) from the tweets as they may not provide useful information for sentiment analysis.
* **Removal of Special Characters**: Remove special characters and punctuation marks from the tweets to focus on the textual content.
* **Removal of Stopwords**: Remove common English stopwords (e.g., 'is', 'the', 'and') as they do not contribute much to sentiment analysis.
* **Tokenization**: Tokenize the tweets into individual words or tokens to facilitate further analysis.
* **Lemmatization**: Lemmatize the tokens to convert them into their base or root form, reducing inflectional forms to a common base form.

**3.3 EXPLORATORY DATA ANALYSIS (EDA)**

After the initial preprocessing of the data, Exploratory Data Analysis (EDA) is conducted to delve deeper into the dataset and understand its characteristics. EDA is a crucial step that helps in uncovering patterns, trends, and anomalies in the data, providing valuable insights for subsequent modeling tasks.

**Visualization of Label Distribution:**

One of the primary tasks during EDA is to visualize the distribution of labels, i.e., hate speech vs. non-hate speech, within the dataset. This involves creating plots such as bar charts or pie charts to illustrate the proportion of each class. Understanding the balance or imbalance between classes is essential as it can impact the model's performance and guide the selection of appropriate evaluation metrics.

**Exploring Most Frequent Words:**

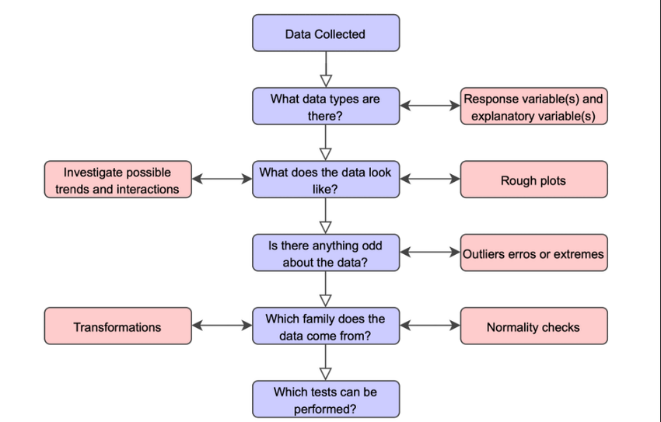
Another key aspect of EDA is to analyze the most frequent words in both categories (hate speech and non-hate speech). This can be achieved by creating word frequency distributions or word clouds, which visually represent the occurrence of words in the dataset. By examining the most common words, researchers can gain insights into the language patterns associated with hate speech and non-hate speech tweets. This analysis can inform feature selection and model training decisions.

**Descriptive Statistics:**

In addition to visualizations, descriptive statistics are calculated to summarize the dataset's key attributes. These statistics may include measures such as mean, median, mode, standard deviation, minimum, maximum, and quartiles for numerical features. For categorical features, frequency counts and percentages may be calculated to understand the distribution of different categories. Descriptive statistics provide a comprehensive overview of the dataset's central tendency, dispersion, and shape, aiding in the identification of outliers and anomalous data points.

**Sentiment Analysis:**

While not explicitly mentioned in the initial description, sentiment analysis could be considered as part of EDA, especially if the dataset includes sentiment labels (positive, neutral, negative) in addition to hate speech labels. Sentiment analysis involves analyzing the sentiment expressed in the tweets and examining its distribution across different classes and categories. This analysis can provide valuable context for understanding the underlying sentiment behind hate speech and non-hate speech tweets, facilitating a deeper understanding of the dataset's content.



**Figure 5:** Exploratory Data Analysis (EDA) Process

**3.4 FEATURE ENGINEERING**

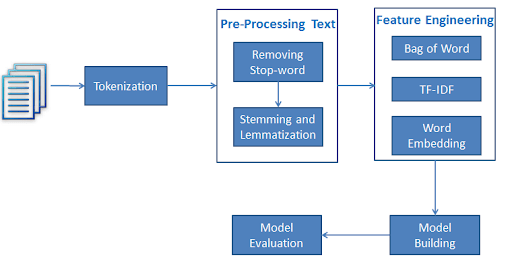
Feature engineering is an essential process in Natural Language Processing (NLP) tasks, where the goal is to transform raw text data into a format that machine learning algorithms can effectively process and learn from. In this project, feature engineering plays a pivotal role in preparing the textual data for classification tasks related to hate speech detection.

**Understanding Feature Engineering in NLP:**

Feature engineering involves extracting meaningful information from text data and representing it in a numerical format. This transformation enables machine learning models to interpret and learn patterns from the data. In NLP, traditional machine learning algorithms require numerical input, which necessitates the conversion of text data into structured features.

**TF-IDF Vectorizer:**

Term Frequency-Inverse Document Frequency (TF-IDF) vectorization is a prominent NLP feature engineering technique. It works by translating text documents into numerical vectors, each of which is a distinct corpus word symbolized by a dimension. Words received weights by the TF-IDF vectorizer according to the significance they are in each document and throughout the corpus.

****

**Figure 6:** TF-IDF Vectorizer in Feature Engineering

**How TF-IDF Works:**

* **Term Frequency (TF)**: This component measures the frequency of a term (word) within a document. Words that occur more frequently within a document are assigned higher TF values.
* **Inverse Document Frequency (IDF)**: IDF measures the rarity of a term across all documents in the corpus. Words that are common across multiple documents receive lower IDF values, while rare words receive higher IDF values.
* **TF-IDF Weighting**: The TF-IDF vectorizer combines TF and IDF to assign weights to words. Words that are frequent within a document but rare across the corpus receive higher TF-IDF weights, indicating their significance in characterizing the document.

**Importance of TF-IDF:**

* **Capture Word Importance**: By incorporating both term frequency and inverse document frequency, TF-IDF effectively captures the importance of words in distinguishing between documents.
* **Dimensionality Reduction**: TF-IDF helps in reducing the dimensionality of the feature space by focusing on informative words while filtering out common stopwords and irrelevant terms.
* **Normalization**: TF-IDF normalization ensures that the resulting feature vectors are robust to variations in document length, enabling fair comparisons between documents of different lengths.

**Benefits for Hate Speech Detection:**

* **Discriminative Features**: TF-IDF generates discriminative features that highlight words specific to hate speech or non-hate speech content, aiding in the classification task.
* **Flexibility**: The TF-IDF approach allows for flexibility in adjusting parameters such as tokenization, n-gram range, and vocabulary size, enabling customization based on the characteristics of the dataset and the requirements of the classification task.

Feature engineering through the TF-IDF vectorizer plays a crucial role in converting raw text data into structured numerical features, facilitating effective machine learning-based hate speech detection. By capturing the importance of words and reducing dimensionality, TF-IDF enables the creation of informative feature representations that enhance the performance of classification models.

**3.5 MODEL BUILDING**

Once the data preprocessing steps are completed and the features are engineered, the focus shifts towards building machine learning models for sentiment analysis, specifically for the task of hate speech detection. In this section, the process of model construction and evaluation is outlined, emphasizing the utilization of a logistic regression classifier.

**Importance of Model Building:**

Model building is a critical phase in the development of any machine learning system, as it involves training algorithms to make predictions or classifications based on the provided input data. For hate speech detection, the goal is to construct a robust model capable of accurately distinguishing between hate speech and non-hate speech content.

**Selection of Logistic Regression Classifier:**

* **Simplicity and Effectiveness**: Logistic regression is chosen as the classification algorithm due to its simplicity and effectiveness, particularly for text classification tasks. It is a linear model that can efficiently handle high-dimensional feature spaces, making it suitable for processing text data represented by TF-IDF vectors.
* **Interpretability**: Logistic regression models provide interpretable results, allowing for the examination of the coefficients associated with each feature, which aids in understanding the contribution of individual words to the classification decision.

**Model Training Process:**

* **Train-Test Split**: A training set and a testing set are the two subsets that collectively make up the dataset. The logistic regression model will be trained on the training set, and its performance is assessed on the testing set.
* **Logistic Regression Model Training:** Using the training set of data, the logistic regression model is trained, where it learns to classify text documents into hate speech or non-hate speech categories based on the engineered TF-IDF features.
* **Optimization**: Hyperparameter tuning may be performed to optimize the performance of the logistic regression model, such as tuning regularization parameters to prevent overfitting or underfitting.

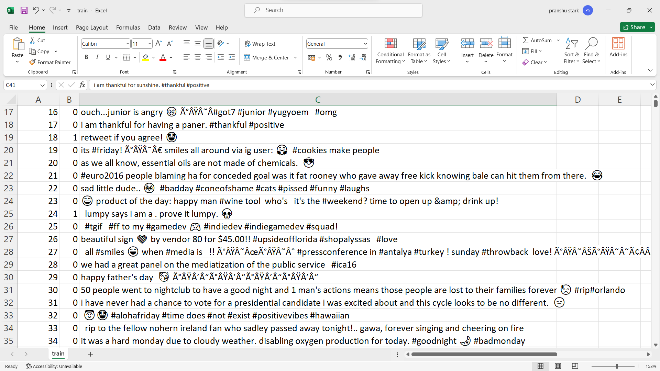
# Proposed methodology

## Dataset

The dataset used in this study was acquired from an external source. It contains tweets related to hate speech or offensive language online along with their corresponding labels. The dataset is split into two CSV files – train.csv and test.csv.

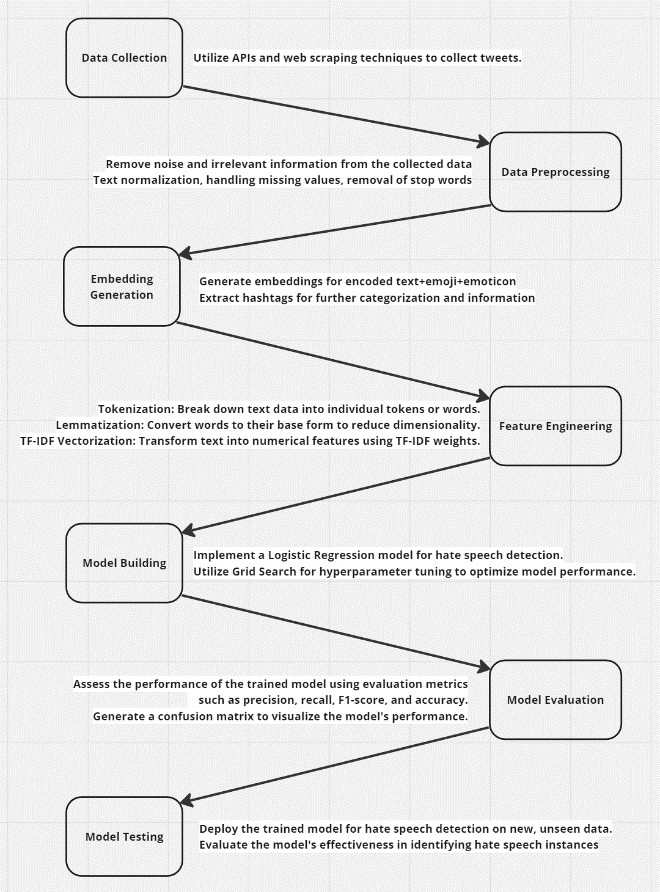
The train.csv file contains 25,000 tweets that are used for training the model. It has three columns – id, label, and tweet. The id column contains a unique identifier for each tweet. The label column contains the class label for the tweet, which is either 0 representing non-hate speech or 1 representing hate speech. The tweet column contains the actual text of the tweet.

Similarly, the test.csv file contains 5000 tweets that are used for testing/evaluating the trained model. It also has the same three columns – id, label, and tweet. The label column in the test file is not used during testing and the model has to predict the labels.



**Figure 7:** Training Dataset (train.csv)

**4.3 PROPOSED ARCHITECTURE**

****

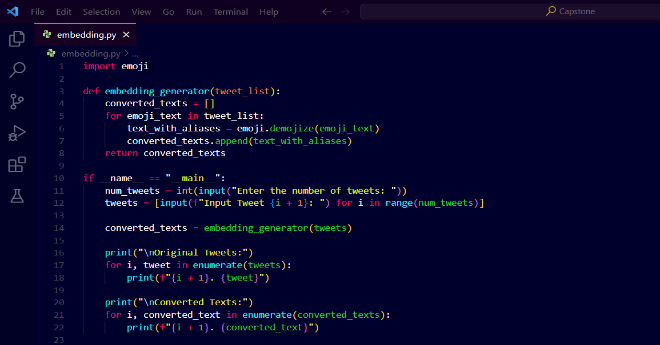
**Figure 14:** Proposed Architecture

**4.3 EMOJI EMBEDDING GENERATION**

In order to incorporate emojis into the sentiment analysis and hate speech detection process, a function named embedding\_generator is implemented. This function processes the tweets containing emojis, generates embeddings, and converts the tweets into text-only while preserving the context conveyed by the emojis. The generated embeddings are then utilized to enhance the existing sentiment analysis models.

The embedding\_generator function takes a list of tweets, tweet\_list, as input. It iterates over each tweet in the tweet\_list and performs the following steps:

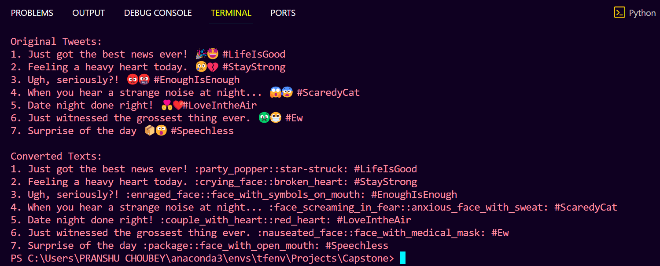
1. **Conversion to Text with Emoji Aliases**: The emoji.demojize() function is applied to each tweet containing emojis, emoji\_text. This function replaces each emoji with a textual representation using emoji aliases. For example, the emoji "😊" may be converted to ":smiling\_face\_with\_smiling\_eyes:".
2. **Building the Converted Texts List**: The converted text with emoji aliases, text\_with\_aliases, is appended to the converted\_texts list. This list contains the processed tweets where emojis are represented as text.
3. **Returning the Converted Texts**: Finally, the converted\_texts list, which now contains the tweets converted to text with emoji aliases, is returned as the output of the embedding\_generator function.



**Figure 15:** Embedding Generation Function

The generated embeddings and converted texts are then utilized in subsequent steps of the sentiment analysis and hate speech detection pipeline. By converting tweets with emojis into text-only format while preserving the emoji context, this functionality enables the incorporation of emojis as valuable features in the sentiment analysis models.

The emoji embeddings derived from the converted texts can capture the emotional connotations and additional sentiment-related information conveyed by emojis. By including these embeddings as features, the sentiment analysis models gain the ability to consider the nuances and emotional cues present in the tweets, enhancing their accuracy and effectiveness in detecting hate speech and analyzing sentiments.



**Figure 16:** Embedding Generation Sample Output

**4.4 HASHTAG EXTRACTION**

To facilitate preprocessing and further analysis, a function named extract\_hashtags is implemented. This function takes in a list of tweets and extracts the hashtags present in each tweet. Additionally, it removes the hashtags from the original tweet text, generating a cleaned version of the tweet. The extracted hashtags and cleaned tweets are returned as a result array.

The extract\_hashtags function takes a list of tweets, tweet\_list, as input. It iterates over each tweet in the tweet\_list and performs the following steps:

1. **Extraction of Hashtags**: Using regular expression re.findall(r'#\w+', tweet), the function identifies hashtags in each tweet. The expression #\w+ matches any word preceded by the "#" symbol. The hashtags are extracted as a list, hashtags.
2. **Removal of Hashtags from the Tweet**: The function uses re.sub(r'#\w+', '', tweet) to remove the hashtags from the original tweet. This regular expression substitution replaces all occurrences of hashtags with an empty string, effectively removing them from the tweet text. The resulting tweet without hashtags is stored in the variable tweet\_without\_hashtags.
3. **Building the Result Array**: The original tweet, the cleaned tweet without hashtags, and the extracted hashtags are appended as a list  to the result\_array. The join() function concatenates the extracted hashtags into a comma-separated string.
4. **Returning the Result Array**: Finally, the result\_array containing the original tweet, cleaned tweet without hashtags, and the extracted hashtags is returned as the output of the extract\_hashtags function.

The extracted hashtags and cleaned tweet texts can be utilized for various purposes, such as preprocessing, analysis, or categorization of tweets based on relevant topics or themes. By separating hashtags from the tweet text, the subsequent preprocessing steps can focus on the remaining content while preserving the information related to hashtags for further analysis.

##### References

1. Chuchu Liu, Fan Fang, Xu Lin, Tie Cai, Xu Tan, Jianguo Liu, Xin Lu - "Improving sentiment analysis accuracy with emoji embedding" in Journal of Safety Science and Resilience, pp. 246-252, 2021, DOI: <https://doi.org/10.1016/j.jnlssr.2021.10.003>
2. Anatoliy Surikov, Evgeniia Egorova - "Alternative method sentiment analysis using emojis and emoticons" in ScienceDirect, pp. 182-193, 2020, DOI: <https://doi.org/10.1016/j.procs.2020.11.020>
3. Ayvaz, Serkan & Shiha, Mohammed. (2017). The Effects of Emoji in Sentiment Analysis. International Journal of Computer and Electrical Engineering. 9. 360-369. 10.17706/IJCEE.2017.9.1.360-369.
4. Vashisht, G., & Jailia, M. (2020). Emoticons & emojis based sentiment analysis: The last two decades. International Journal of Scientific and Technology Research, 9(3), 6398-6405.
5. Guibon, G., Ochs, M., & Bellot, P. (2016, June). From emojis to sentiment analysis. In WACAI 2016.
6. Wolny, W. (2016, June). Emotion analysis of Twitter data that use emoticons and emoji ideograms. In Proceedings of the 25th International Conference on Information Systems Development (ISD2016 KATOWICE) (pp. 1-6). University of Economics in Katowice.
7. Jagadishwari, V., Indulekha, A., Raghu, K., & Harshini, P. (2021, November). Sentiment analysis of social media text-emoticon post with machine learning models contribution title. In Journal of Physics: Conference Series (Vol. 2070, No. 1, p. 012079). IOP Publishing. DOI 10.1088/1742-6596/2070/1/012079
8. Bhaskaran, S., Veeramanickam, M. R. M., Hariharan, S., Bharathiraja, N., Pradeepa, K., & Marappan, R. (2022, December). Sentiment Analysis Model using Text and Emoticons for Pharmaceutical & Healthcare Industries.