1. Introduction and background

The use of social media and social networking has been rising over time as a result of an increase in internet usage all over the world. Social media platforms and mobile technologies are two significant recent evolutions. This has had a profound impact on how individuals communicate with one another. Numerous social media sites, such as Snapchat, Facebook, Instagram, Twitter, and others, have expanded into global networks. These social media platforms provide an enormous quantity of material and information because people use them daily to interact, voice their ideas, and share their perspectives. Although a large portion of this kind of virtual contact is harmful, it is also quite helpful and constructive. Threats, insults, and other components of cyberbullying can occasionally be found on social media platforms. Many people use online social networks. As a result, protecting network users against anti-social conduct is an essential occupation. In this project we are aiming to develop a method in order to recognize and report any toxic language, insults and threats in social network. Every day, an enormous amount of information is generated via online conversations since these platforms allow individuals to debate, express themselves, and voice their opinions. This is a highly productive condition that has the potential to greatly improve human life quality, but it also has a high risk of destruction. However, these discussions frequently have a dark side and can lead to arguments on social media when one side may use derogatory language or poisonous remarks. Additionally, certain sites are frequently compelled to either limit user comments or shut them down entirely due to hate speech, threats and toxic language. In the realm of online communication, numerous Machine Learning models have been conceived and employed to filter out inappropriate language and protect internet users from the perils of cyberbullying and harassment. The application of advanced techniques, including machine learning, deep learning, and others, can be considered for the differentiation between toxic and non-toxic comments. However, this project will prioritize the utilization of basic methods to address this issue. Emphasis will be placed on the maintenance of accessibility and ease of implementation, rendering it well-suited for the project's scope and the level of expertise at hand. The primary reliance will be on straightforward methods, which will be implemented using Python to filter and process the data.

2. Problem description

As mentioned above social media had become a part in our daily life, unfortunately Social media platforms can occasionally contain threats, insults, and other elements of cyberbullying. The issue of identifying toxic comments and threats on social media was examined and in this project a data will be collected and differentiate toxic classified comments from non-toxic comments.

2.1 Data

We will use a manageable dataset of online comments and messages collected from social media and discussion forums. The dataset will be curated to contain examples of both toxic and non-toxic text.

2.2 Target of the Study

The primary objective is to develop a rule-based system by python, that can effectively classify text as either "Toxic" or "Non-Toxic." Toxicity in this context encompasses negative language, hate speech, harassment, and threats.

3. Proposed Solution

Our proposed solution involves the development of a rule-based system for the identification and reporting of toxic language in social networks. This system will be implemented using Python for efficient text processing and classification.

3.1 Data Processing

To prepare the data for classification, we will conduct essential data processing steps, including text tokenization, lowercasing, stop word removal, and stemming or lemmatization.

3.2 Defining Toxic Elements

We will compile a list of toxic words, phrases, and patterns that encompass negative language, hate speech, insults, harassment, and threats.

3.3 Rule-Based Classification

The system will use these predefined rules and patterns, along with the preprocessed data, to classify text as either "Toxic" or "Non-Toxic." A comment will be categorized as "Toxic" if it contains elements from our list.

3.4 Python Implementation

Python, known for its versatility and extensive libraries, will be the language of choice for system implementation.

3.5 Model Training and Testing

While the primary approach is rule-based, we will conduct model training and testing with labeled data to enhance system performance.

3.6 Performance Evaluation

The system's performance will be assessed with a focus on minimizing false positives and false negatives. This analysis will guide ongoing improvements.

This solution combines data processing, rule-based classification, and Python implementation to create a practical tool for recognizing and reporting toxic language in social networks, ensuring a safer online environment for users.

4. Plan for testing and result analysis.

4.1 Data Simulation and Preparation

-Given that this is a project simulation, we will create a synthetic dataset that mirrors online comments and messages with a mix of toxic and non-toxic text.

-Data preprocessing steps, including tokenization, lowercasing, stop word removal, and stemming or lemmatization, will be applied to prepare the synthetic dataset for analysis.

4.2 Test Scenarios

-We will define test scenarios that represent a range of online interactions, including diverse comment types and a variety of simulated toxic language, hate speech, insults, and threats.

-A representative selection of comments from the synthetic dataset will be chosen for testing to ensure a balanced distribution of toxic and non-toxic content.

4.3 Performance Metrics

-For result analysis, we will use performance metrics such as precision, recall, F1 score, and accuracy. These metrics will help assess the effectiveness of the simulated system's rule-based classification.

-Precision will measure the system's ability to correctly identify toxic comments, while recall will assess its capacity to detect all simulated toxic comments.

-The F1 score combines precision and recall for a balanced assessment, and accuracy measures the overall correctness of classifications.

4.4 Model Training and Testing

-Similar to real-world scenarios, we will iterate through multiple training cycles using the synthetic data to refine the simulated system's rule-based classification.

-The objective is to continuously assess and fine-tune the system to improve its ability to accurately distinguish between simulated toxic and non-toxic comments.

4.5 False Positive and False Negative Analysis

-A critical part of result analysis will focus on understanding and mitigating false positives (correct comments classified as toxic) and false negatives (simulated toxic comments not correctly classified).

-This analysis will guide system enhancements and rule adjustments in the simulated environment.