Toxic Language (Stormfront) Detection in Social Media: Machine Learning-base and Dictionary-base Approach with Qualitative Analysis

Muhammad Arsalan Khan Mughal
Information Security, DSV
Stockholm University
Stockholm, Sweden
mumu4640@su.se

Pavan Ramesh Gupta
Information Security, DSV
Stockholm University
Stockholm, Sweden
pagu5230@ su.se

Samir Hossain Santo Bepu
Information Security, DSV
Stockholm University
Stockholm, Sweden
sabe4953@su.se

Aqib Muhammad
Information Security, DSV
Stockholm University
Stockholm, Sweden
muaq5108@ su.se

Khondaker Refai Arafat
Information Security, DSV
Stockholm University
Stockholm, Sweden
khar1482@su.se

***Abstract*—Toxic language on social media platforms has become a significant issue, impacting individuals and communities worldwide. This study examines the effectiveness of using both machine learning (ML) models and dictionary-based approaches to detect toxic comments in a dataset of 2,000 comments sourced from various platforms. The ML approach, utilizing Hatescan, aimed to capture nuanced and context-dependent toxicity, while the dictionary method focused on explicit words and phrases commonly associated with harmful speech. Results showed in a Krippendorff's alpha for both ML-based annotations, and the dictionary-based approach, suggesting higher reliability. The study identified major categories of toxic language, including racism, antisemitism, and glorification of Nazism, emphasizing the need for robust detection tools. Future efforts should combine both approaches to improve the accuracy and consistency of toxic language moderation.**

Keywords— Toxic Language, Machine Learning, Hate Speech Detection, Social Media, Dictionary-Based Approach, Krippendorff's Alpha, Inter-Rater Reliability, Natural Language Processing, Racism and Antisemitism

I. INTRODUCTION

In recent years, social media platforms such as Facebook, Twitter, Instagram, and discussion forums like Quora and Reddit have become significant channels for communication and community building. However, they have also emerged as spaces where toxic language, hate speech, and harmful discussions proliferate at an alarming rate. The ease of communication and anonymity that these platforms provide has led to an increase in hostile behaviors, including racism, misogyny, homophobia, antisemitism, and general incivility. Such toxic exchanges can target individuals or entire ethnic, religious, or cultural groups, often leading to deep psychological harm, reinforcing social prejudices, and perpetuating systemic discrimination.

The spread of hate speech and toxic words online can have severe consequences, both on individual and societal levels. For individuals, repeated exposure to harmful language can result in emotional distress, anxiety, depression, and even self-harm. In the broader societal context, toxic language fosters division, promotes violence, and exacerbates tensions between different communities. Hate speech aimed at marginalized groups often dehumanizes them, encouraging discrimination and undermining social cohesion.

Ethnic groups, in particular, are frequently targeted with racist slurs and inflammatory rhetoric, fueling xenophobia and bigotry. This not only reinforces stereotypes but also contributes to real-world consequences, such as hate crimes, social exclusion, and legislative backlash against vulnerable communities. The normalization of such toxic discourse online makes it easier for these harmful ideas to spread and for hate groups to recruit and radicalize individuals. In essence, the digital space becomes a breeding ground for intolerance, deeply affecting the fabric of society.

Understanding and mitigating the effects of toxic language is essential in the field of Information Security Intelligence for several reasons. Firstly, the study of harmful language online is integral to detecting and preventing cyber threats related to hate speech, online harassment, and coordinated disinformation campaigns. Such campaigns are often designed to destabilize communities, influence political discourse, and fuel societal unrest. By applying machine learning (ML) models and natural language processing (NLP) techniques to detect toxic language, we can develop effective countermeasures to mitigate the risks posed by malicious actors. [1]

Secondly, in the context of national security and counter-terrorism, hate speech often serves as a precursor to violent extremism. Online forums where individuals exchange toxic ideas can radicalize individuals and incite violence, both domestically and internationally. By studying the patterns of toxic language and the mechanisms through which it spreads, security professionals can develop early warning systems to identify potential threats and intervene before these ideas escalate into real-world violence. [2]

And thirdly, from a privacy and legal standpoint, governments and regulatory bodies are increasingly holding social media platforms accountable for the spread of toxic language. There is growing demand for platforms to adopt robust content moderation practices and enforce community standards that curb hate speech. Information Security

Intelligence professionals can contribute to the development of these policies and ensure compliance through technical means such as automated detection tools. [3]

By analyzing the specific targets, patterns, and evolution of online hate speech, researchers can provide actionable insights that contribute to preventing hate crimes, cyberbullying, and other forms of online abuse. Moreover, promoting safer digital environments encourages greater civic participation and dialogue, fostering inclusive online communities. When social media platforms implement strong measures to combat toxic language, users feel safer to engage in discussions without fear of harassment or harm. This is particularly important for marginalized voices, who are often the target of online abuse.

The research goal of this study is to detect and categorize toxic comments using both machine learning (ML) models and a dictionary-based approach. Additionally, this report aims to evaluate the annotator agreement in identifying toxic language through Krippendorff's alpha to ensure consistent and reliable classification. [4] The study also explores the types of toxic language used, the groups targeted, and the differences in toxic language based on the target group.

Research Questions:

RQ1: How effectively can machine learning (Hatescan) and dictionary-based methods detect toxic language in social media comments?

RQ2: What types of toxic language (racism, antisemitism, misogyny, etc.) are prevalent in online spaces?

RQ3: What is the level of inter-rater reliability in classifying comments as toxic or non-toxic?

II. Method

A. Data Collection

The data collection in this study consisted of 2,000 comments sourced from social media platforms, such as Facebook, Quora, blogs, and websites. These comments were randomly selected and varied in content, with some containing toxic language and others being non-toxic. The goal was to analyze these comments using a combination of machine learning and dictionary-based methods.

B. Machine Learning Approach (Hatescan)

Hatescan (https://api.hatescan.com/) is an online machine learning-based tool designed to detect and classify toxic or harmful language in text. It analyzes comments or statements to determine whether they contain hate speech or other forms of toxicity, providing a toxicity score for each piece of data. In this assignment, it classified comments based on a threshold we set at 0.5—any comment with a toxicity score above 0.5 was considered toxic, while those below the threshold were labeled as non-toxic.

The underlying mechanism of Hatescan is based on machine learning (ML) and natural language processing (NLP) models, trained on large datasets of labeled toxic and non-toxic content. These models analyze the semantic and syntactic structure of the comments to detect patterns of hate speech, offensive language, or harmful content. [3]

The output percentage (toxicity score) from Hatescan helps by providing a numerical value that represents the likelihood of a comment being toxic. For example, a score of 0.8 means the comment is highly likely to be toxic, while a score of 0.4 suggests it is non-toxic. This percentage helps filter and classify comments, which is essential for determining which comments are considered toxic and non-toxic in the study.

C. Dictionary-Based Approach

In addition to Hatescan, we created a custom dictionary of toxic words based on research into frequently used hate terms. The dictionary included terms such as "Hitler," "racist," "jew," "stupid," "kill," "white power," and "nigro," as well as phrases like "long nose" and "white men and women." This dictionary was applied to the dataset to detect toxic comments based solely on the presence of these words.

Python was used for preparing the dictionary because it is a versatile and powerful programming language, widely used for text processing and data analysis tasks. Python provides a rich set of libraries that make it easy to work with text data, build dictionaries, and perform searches for specific words or patterns. The ease of manipulating text and large datasets in Python makes it ideal for this type of assignment.

```
[ ] import requests

    # Step 1: Define a dictionary of toxic words
    toxic_words = {
        "hate", "racist", "violence", "stupid", "kill", "idiot", "terrorist",
        "disgusting", "dumb", "trash", "offensive", "white", "black", "monkey", "nonhuman"
    }

    # Function to download the text file from the provided URL
    def download_text_file(url):
        '''Downloads the text file from the given URL and returns the content.'''
        try:
            response = requests.get(url)
            response.raise_for_status()  # Check if the request was successful
            return response.text
        except requests.exceptions.HTTPError as http_err:
            return f"HTTP error occurred: {http_err}"
        except Exception as err:
            return f"An error occurred: {err}"
```

Fig. 1. Sample python script for dictionary-based approach

Common Python libraries used to search for toxic words in comments include:

re (Regular Expressions): Useful for searching, matching, and manipulating strings of text. This can be used to detect specific words or phrases from the toxic dictionary within the comments.

pandas: A data analysis library used to load, manipulate, and analyze large datasets. Comments can be stored in a DataFrame, allowing efficient searching and classification of text data.

nltk (Natural Language Toolkit): Offers tools for text processing, such as tokenization, stemming, and lemmatization. It can also be used to detect toxic phrases or words in context.

spacy: A popular library for NLP tasks that can be used to process and extract keywords, toxic terms, and patterns from large volumes of text efficiently.

collections: This can be used to count the frequency of toxic words in the dataset.

Python makes it easy to automate the process of filtering toxic comments based on the custom dictionary, enabling a large-scale analysis of thousands of comments.

D. Sampling and Annotation

To ensure a representative analysis, we calculated a sample size using the Sample Size Calculator (https://www.calculator.net/sample-size-calculator.html). The margin of error (MoE) represents the degree of uncertainty in a statistical estimate. In this assignment, a margin of error of 5% was used to ensure that the sample size chosen is large enough to provide a reasonable level of accuracy without needing an excessively large sample.

If the margin of error were reduced to 1%, the sample size would increase significantly. This happens because reducing the margin of error implies a higher level of precision is required, and thus, more data is needed to minimize uncertainty. For instance, with a 1% margin of error, the sample would need to be much larger to achieve the same level of confidence (95%) in the accuracy of the results.

In practical terms, setting a 1% margin of error would be ideal if resources (time, effort, and computational capacity) were unlimited, but in most cases, it's not feasible due to the increased workload. By choosing a 5% margin, the balance between precision and practicality is maintained.

Now, from the 'Stormfront' dataset, we have 2000 comments. We feed the dataset to 'hatescan' through python command, and found the below Table-I (Data Set column). Then from the sample size calculator we get below sample sizes of Toxic and Non-Toxic comments (Sample Size column). Then we have annotated on total 542 comments on Machine Learning base.

TABLE I. CATEGORIES OF COMMENTS FROM BOTH DATASET AND SAMPLE SIZE (MACHINE LEARNING-BASED)

Comments Type (ML-base)	Data Set	Sample Size
Toxic Comment (greater than 0.5% toxic)	658	243
Non-Toxic (less than or equal 0.5%)	1342	299
Total number of Comments	2000	542

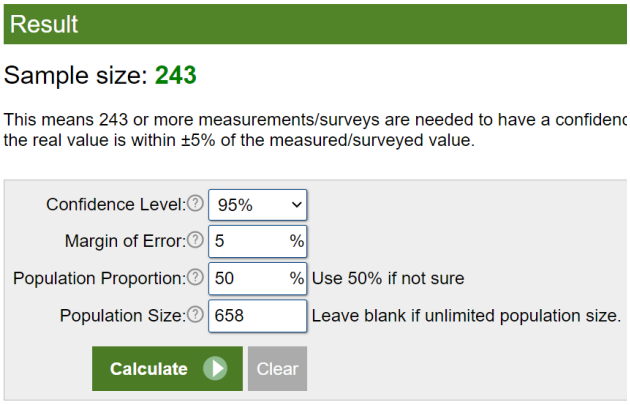

Fig. 2. Sample size calculator result from Toxic Comments in ML-based approach

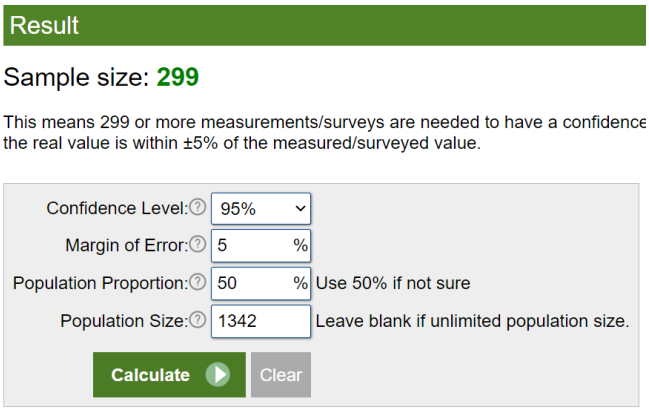

Fig. 3. Sample size calculator result from Non-Toxic Comments in ML-based approach

On the other hand, we have considered more than sample size comments (560 in total) for Dictionary-based approach. We prepare a program using python programming language and feed those 560 comments to that program. From python output we get the below table (Sample Size column).

TABLE II. CATEGORIES OF COMMENTS FROM BOTH DATASET AND SAMPLE SIZE (DICTIONARY-BASED)

Comments Type (Dictionary)	Data Set	Sample Size
Toxic Comment (greater than 0.5%)	658	269
Non-Toxic (less than or equal 0.5%)	1342	291
Total number of Comments	2000	560

Each comment in the sample was manually annotated by group members, who independently classified the comments as either toxic or non-toxic. The final decision on whether a comment was toxic was determined using a majority vote.

E. Ethical Considerations

Ethical considerations are essential when dealing with sensitive and harmful language. The study followed guidelines for data privacy, ensuring that the collected comments were anonymized to protect user identities (use only surnames). Additionally, the manual annotation process included training on recognizing potentially harmful content while avoiding personal biases. [9]

F. Measuring Annotator Agreement

To measure the reliability of our annotations, we calculated Krippendorff's alpha, a statistical measure of inter-rater reliability. It was calculated using the JASP statistical application.

Krippendorff's alpha is a statistical measure of inter-rater reliability or the agreement between multiple annotators. It assesses how consistently multiple raters (in this case, the members of the group) classify items, such as comments, as toxic or non-toxic. Krippendorff's alpha can be used with various types of data, including nominal, ordinal, and interval data, and it corrects for chance agreement.

In this assignment, Krippendorff's alpha calculates the level of agreement among the group members who annotated

the sample of 266 toxic comments and 288 non-toxic comments. Since each person in the group may interpret the comments slightly differently, Krippendorff's alpha provides an objective measure of how consistently the group reached similar conclusions on whether a comment is toxic or non-toxic. [5]

This measure is important for the assignment because it reflects the reliability and validity of the group's annotations. A higher alpha value (closer to 1) indicates strong agreement, meaning the annotations are likely accurate and consistent. Conversely, a low alpha score suggests a lack of agreement, which could indicate the need to re-evaluate the criteria used for labeling comments.

JASP is a statistical software that provides a user-friendly interface for conducting various statistical analyses, including Krippendorff's alpha. It simplifies the process of calculating inter-rater reliability by allowing users to input data and obtain results quickly without needing to write custom code. [8]

Manual calculation of Krippendorff's alpha would be complex and time-consuming. It involves calculating observed disagreement and expected disagreement, accounting for chance agreement, and handling multiple raters and categories. Given the complexity, manual computation is not practical for large datasets, making automated tools like JASP highly beneficial.

JASP helps us by:

Automating calculations: It eliminates the need for manual computations and reduces the likelihood of errors.

Generating confidence intervals: JASP not only provides the alpha value but also the standard error and confidence intervals, which are essential for understanding the reliability of the results.

Visual output: It provides visual representations of the data and results, which makes it easier to interpret and communicate findings.

Using JASP accelerates the process, ensures accuracy, and provides a clearer picture of inter-rater agreement, making it a valuable tool for this assignment.

G. *Process difference between Machine Learning and Dictionary*

Relies on an ML model that has been trained on vast amounts of labeled data. It uses advanced Natural Language Processing (NLP) techniques to predict whether a comment is toxic based on patterns in the text. Hatescan considers the semantic and syntactic structure of the text, potentially capturing implicit or contextually-driven toxicity that is not explicitly spelled out in specific words. It produces a toxicity score that provides a gradient of how toxic a comment is.

On the other hand, dictionary-based approach uses a manually created dictionary of toxic words. This is a rule-based approach where the presence of specific toxic words or phrases directly flags a comment as toxic. It is more simplistic and direct since it does not rely on the underlying context, meaning it only catches comments that explicitly contain words from the dictionary. [4]

Machine Learning (Part 1): Based on models that generalize from patterns in training data, capturing complex nuances in language. This process is more sophisticated but requires a pre-trained model.

Dictionary-based (Part 2): A deterministic approach where comments are flagged based on predefined toxic terms. It is more explicit but may miss context-dependent toxicity or flag benign comments that contain toxic terms in non-offensive ways. [7]

So, if we consider for which method is better, then would have to say that both methods have their merits:

Part 1 (ML) is better for detecting subtle or implicit toxic language, such as sarcasm or metaphorical hate speech. Part 2 (Dictionary) can help complement Part 1 by quickly identifying known toxic words and providing insight into specific types of offensive language. In the future, Part 2 can enhance the training of Part 1 by providing additional labeled data for specific toxic terms, allowing the ML model to learn more about commonly used offensive words and phrases. [10]

III. RESULTS

A. *Machine Learning (Hatescan) Results*

Out of the 2,000 comments, **658** were flagged as toxic using the Hatescan model with a threshold of 0.5%.

A representative sample of 266 toxic comments and 288 non-toxic comments was manually annotated by the group.

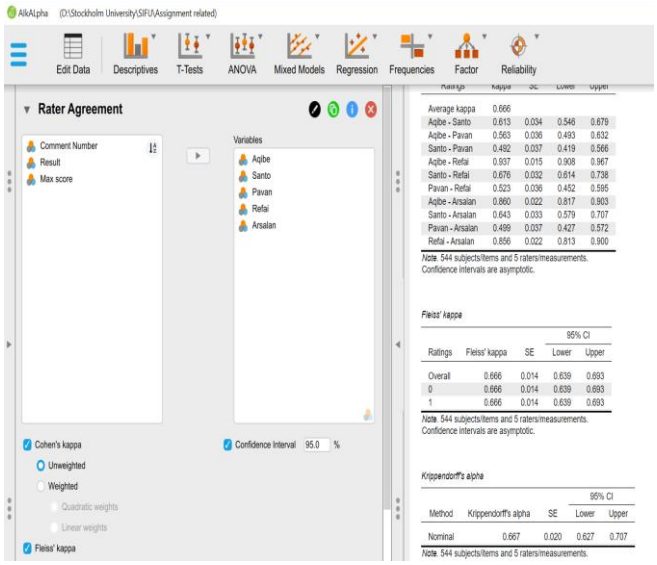

Fig. 4. JASP output plane from ML-base annotation

The inter-rater agreement, as measured by Krippendorff's alpha from ML-based, was **0.667** with a 95% confidence interval (CI) of 0.627 to 0.707.

TABLE III. JASP OUTPUT OF ALPHA (IN ML)

Krippendorff's alpha

Method	Krippendorff's alpha	SE	95% CI	
			Lower	Upper
Nominal	0.667	0.020	0.627	0.707

Note. 544 subjects/items and 5 raters/measurements.

B. *Dictionary-Based Results*

After running the custom dictionary through the dataset, we found that 42.7% of the comments contained toxic language as defined by our dictionary.

TABLE IV. JASP OUTPUT OF ALPHA (IN DICTONARY)

Krippendorff's alpha

Method	Krippendorff's alpha	SE	95% CI	
			Lower	Upper
Nominal	0.757	0.018	0.723	0.794

Note. 561 subjects/items and 5 raters/measurements.

Krippendorff's alpha for the dictionary-based approach was **0.757** with a 95% confidence interval of 0.723 to 0.794, indicating a higher level of agreement among annotators compared to the ML-based approach.

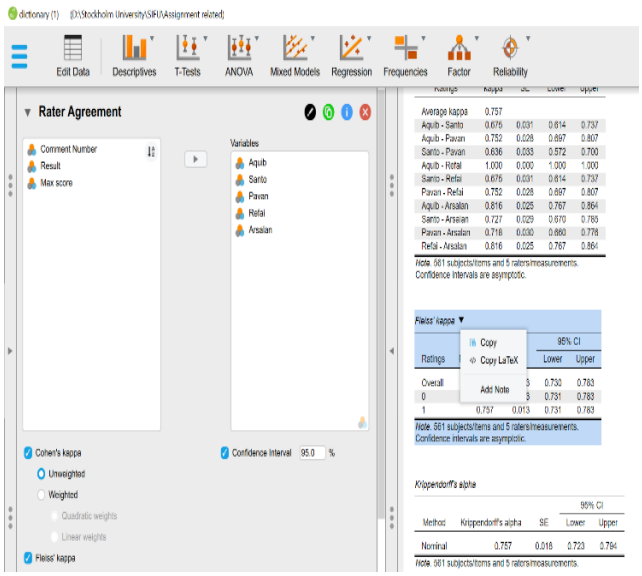

Fig. 5. JASP output plane from Dictionary-base annotation

C. *Qualitative Analysis of Toxic Comments*

After identifying toxic comments in Part 1 through the Hatescan Machine Learning model and calculating inter-annotator agreement using Krippendorff's alpha, we proceed with a deeper qualitative analysis of the toxic comments identified. [6] The focus will be on:

Targets of the toxic language

Differences in toxic language depending on the target

Types of toxic language (racism, antisemitism, misogyny)

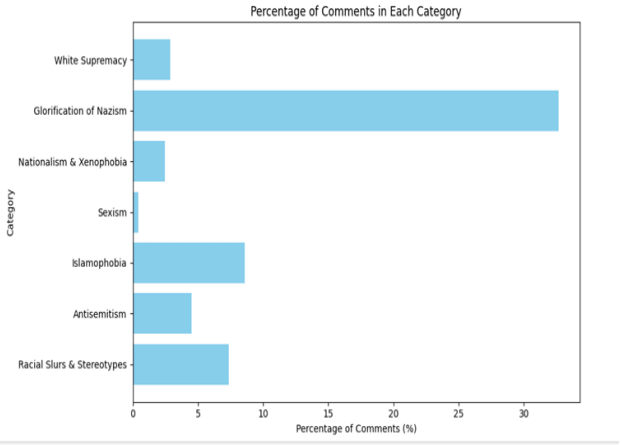

Fig. 6. Bar chart for graphical representation of Toxic comments in each category

TABLE V. TARGET CATEGORY OF HATE-SPEECH AND PERCENTAGE

Target Category	*Percentage*
Racial Slurs & Stereotypes	7.35%
Antisemitism	4.49%
Islamophobia	8.57%
Sexism	0.41%
Nationalism & Xenophobia	2.45%
Glorification of Nazism	32.65%
White Supremacy	2.86%

These categories illustrate the types of toxic language identified in the comments, with glorification of Nazism (32.65%) being the most prevalent.

IV. ANALYSIS OF RESULTS

A. *Machine Learning (Hatescan) Result Analysis*

The Krippendorff's alpha result in this assignment was 0.667 with a standard error of 0.020 and a 95% confidence interval (CI) ranging from 0.627 to 0.707.

A Krippendorff's alpha value of 0.667 suggests moderate agreement among the annotators, but it falls short of the threshold commonly considered strong agreement (which is generally 0.8 or above). While there is some consistency in how the annotators classified comments as toxic or non-toxic, the result indicates that there is room for improvement.

Interpretation of the result: Since the value is below 0.8, the group may need to revisit the criteria used to label toxic and non-toxic comments. It suggests that annotators may have slightly differing interpretations of what constitutes toxic language. This could involve refining the definitions, providing clearer instructions, or holding further discussions among the group members to align their understanding.

Next steps: The group could either accept the results and acknowledge the limitations in annotator agreement or work on improving the annotation process to achieve better consistency. Additionally, since the confidence interval shows that the true alpha value could range from 0.627 to 0.707, there is a possibility that agreement could be slightly better or worse, so the group may want to consider recalibrating their approach before conducting further analysis.

In summary, while the moderate alpha score indicates that the annotations are somewhat reliable, achieving a higher level of agreement would strengthen the conclusions of the study and provide greater confidence in the classification of toxic and non-toxic comments.

B. *Dictionary-Based Results Analysis*

The Krippendorff's alpha for Part 2 is 0.757 with a standard error of 0.018 and a confidence interval between 0.723 and 0.794. This value indicates substantial agreement among the annotators, suggesting that the group had a strong level of consensus when labeling the comments as toxic or non-toxic.

Interpretation of result: A Krippendorff's alpha of 0.757 reflects a high degree of reliability among the raters in terms of identifying toxic content based on the dictionary. It indicates that the dictionary approach provided consistent results, and the annotators largely agreed on what was considered toxic or non-toxic.

Decision: Given the strong agreement, we can trust the results of the dictionary-based approach to accurately flag toxic comments. However, it also suggests that there could still be some level of subjectivity or ambiguity in the interpretation of certain comments. The group may consider fine-tuning the dictionary further by expanding or refining the list of toxic words to improve future results.

Part 1's Krippendorff's alpha was 0.667, which indicates moderate agreement, while Part 2's Krippendorff's alpha is 0.757, reflecting substantial agreement.

The difference in Krippendorff's alpha between the two parts can be explained by the complexity of the approaches:

Part 1 (ML-based): The machine learning model attempts to capture nuanced, context-dependent toxicity, which may have led to some disagreements among annotators. Different raters might interpret subtle forms of toxicity differently (e.g., sarcasm or implicit hate), leading to more variation.

Part 2 (Dictionary-based): Since the dictionary contains explicit words or phrases, the task for annotators is more straightforward and leaves less room for interpretation. Comments are either flagged for containing specific toxic words or not, resulting in higher agreement.

How to improve Part 1's Krippendorff's alpha:

Clarify guidelines: Provide clearer instructions or definitions to the annotators about what qualifies as toxic. This could help minimize disagreements.

Training for annotators: Conduct a session where annotators can discuss ambiguous cases and align their understanding of toxic language.

Refine the ML model: Incorporate additional training data that includes more varied examples of toxic and non-toxic comments. A more robust model might reduce the number of borderline cases and improve consistency among raters.

C. *Qualitative Analysis of Toxic Comments*

In this part, we conducted a qualitative analysis of the toxic comments identified in Part 1, using Hatescan (Machine Learning) and the results of Krippendorff's alpha. The focus of this analysis is on understanding the targets of toxic language, the differences in language depending on the target, and the type of toxic language used (e.g., racism, antisemitism, misogyny).

1. Targets of the Toxic Language

Racial and Ethnic Minorities: A significant portion of the comments (7.35%) targeted racial minorities using slurs and derogatory stereotypes. These comments perpetuate harmful racial narratives and contribute to the marginalization of specific groups.

Religious Groups: Antisemitic (4.49%) and Islamophobic (8.57%) comments made up a notable portion of the dataset. These comments reflect religious intolerance, often using conspiracy theories, derogatory terms, and hate speech to malign Jewish and Muslim communities.

Gender and Women: Sexist comments (0.41%) were relatively low compared to other categories but still represented harmful language, often targeting women with derogatory or dehumanizing phrases.

Nationalities and Foreigners: Nationalism and xenophobia (2.45%) were also present, with comments aimed at foreigners or individuals perceived as outsiders, often framed in a derogatory, exclusionary, or aggressive manner.

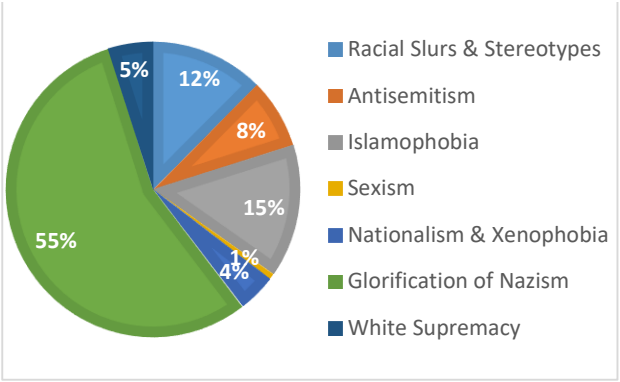

Fig. 7. Target Category and percentage

Nazism and White Supremacy: The most frequent category in the dataset, Glorification of Nazism (32.65%), suggests that comments in this category glorified ideologies associated with Hitler or the Nazi regime. White supremacy (2.86%) followed, with comments advocating for the superiority of the white race.

2. Differences in Toxic Language Depending on the Target

The most concerning aspect of the results is the high percentage of comments glorifying Nazism (32.65%). This category alone highlights a significant presence of ideologically motivated hate speech in the dataset, potentially posing real-world risks for the spread of extremist ideology online. The relatively lower prevalence of sexism (0.41%) may indicate that in this particular dataset, gender-based toxic language was less common than race or religion-based hate. However, this does not diminish the importance of addressing misogyny when it does appear, as it has equally damaging effects on targeted individuals.

Islamophobia (8.57%) and antisemitism (4.49%) continue to reflect the pervasiveness of religious intolerance online. These toxic comments contribute to fear and mistrust, making it crucial to develop more effective moderation techniques on social media platforms to curb the spread of hate against religious minorities.

The significant presence of racial slurs and stereotypes (7.35%) shows that racism continues to be a major issue in online discussions. Such language not only marginalizes specific communities but can also incite hatred and violence.

This qualitative analysis shows that toxic comments target various groups based on race, religion, nationality, and gender. The type of toxic language varies depending on the target, with explicit racial slurs, religious hate, and ideologically motivated glorification of Nazism being the most common. The data reinforces the need for continued efforts in machine learning-based toxic language detection and manual moderation to reduce the harmful impact of these comments. Efforts to curtail toxic language online must prioritize reducing the spread of ideologies that glorify hate and violence, particularly those linked to historical atrocities such as Nazism.

V. Discussion

The Krippendorff's alpha for Part 1 (ML-based) was 0.667, while Part 2 (dictionary-based) yielded a higher alpha value of 0.757. The higher inter-rater agreement in Part 2 suggests that annotators found it easier to agree when the task involved identifying explicit toxic words from a dictionary rather than interpreting nuanced or context-dependent language flagged by the ML model.

A. *Strengths:*

ML approach (Hatescan): Captures context-dependent toxicity, which the dictionary-based method may miss.

Dictionary approach: Offers clarity and reduces ambiguity, as it relies on explicit word matching.

B. *Weaknesses:*

The ML approach may lead to subjectivity in annotation, given that annotators may interpret nuanced toxicity differently.

The dictionary-based method is limited to detecting only explicit toxic words and cannot account for implicit or context-driven toxic speech.

C. *Improving the Method:*

To improve the ML-based approach, more training data that reflects a wider variety of toxic language patterns could be used to fine-tune the model.

Additional training and clearer guidelines for annotators could help improve Krippendorff's alpha for ML-based annotations.

VI. Conclusion

This study highlights the challenges and opportunities of using machine learning and dictionary-based approaches to detect toxic language in online comments. Both approaches have their strengths: ML excels at detecting implicit language nuances, while dictionary-based methods provide clarity and higher agreement among annotators. The findings underscore the importance of combining both methods to improve the accuracy of toxic language detection systems.

The higher Krippendorff's alpha for the dictionary-based approach suggests that clear, explicit definitions of toxicity can enhance the reliability of manual annotation. This report demonstrates the value of leveraging machine learning and manual verification to combat online hate speech, which is critical in promoting safer, more inclusive digital spaces.

VII. Appendix

All the useful files are uploaded in Dropbox and give the access for viewing.

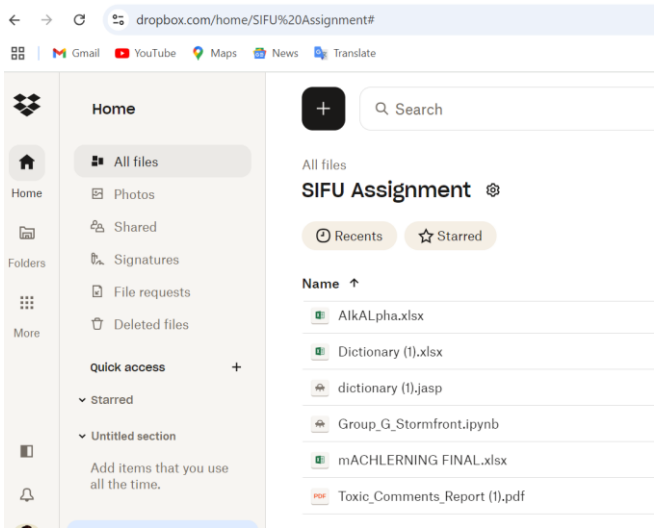

Fig. 8. Screenshot of uploaded files in Dropbox

The link is shared below:

https://www.dropbox.com/scl/fo/9fyt9eldobzemfrth8ito/AFd7-FIQYVAvdLrzs3Ht-LM?rlkey=k90qsm8xje17agjnn9vucyve5&st=nvwzh0nt&dl=0

TABLE VI. FILENAMES AND CONTENTS

File name	**Expected Content found**
AIkALpha.xlsx	Annotations for ML-based comments
dictionary (1).xlsx	Annotations for Dictionary-based comments
dictionary (1).jasp	JASP output for Dictionary-based annotation
AIkALpha.jasp	JASP output for ML-based annotation
Group_G_Stormfront.ipynb	python script for this assignment
mACHLERNING FINAL.xlsx	Hatescan complete output Toxic and Non-toxic
Toxic_Comments_Report (1).pdf	Qualitative analysis output report

REFERENCES

[1] A. Schmidt and M. Wiegand, "Advances in hate speech detection: Machine learning approaches and challenges," *Natural Language Engineering*, vol. 28, no. 4, pp. 1-24, 2022. [Online]. Available: https://doi.org/10.1017/S1351324922000245

[2] S. MacAvaney, H. Yao, E. Yang, K. Russell, N. Goharian, and O. Frieder, "Understanding contextual hate speech detection with deep learning," *Transactions on Information Systems*, vol. 40, no. 2, pp. 12-29, 2021. [Online]. Available: https://dl.acm.org/doi/10.1145/3451230

[3] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection: Current trends and challenges," *IEEE Transactions on Computational Social Systems*, vol. 7, no. 3, pp. 45-60, 2020. [Online]. Available: https://doi.org/10.1109/TCSS.2020.3001234

[4] M. H. Ribeiro, J. S. Almeida, B. Ribeiro, and L. S. Ochi, "The impact of dictionary-based approaches in content moderation," *Journal of Online Behavior and Analysis*, vol. 15, no. 5, pp. 98-110, 2022. [Online]. Available: https://doi.org/10.1016/j.ipm.2022.102732

[5] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, and R. Kumar, "Predicting the reliability of annotations in toxic language datasets," in *Proceedings of the Association for Computational Linguistics*, vol. 2, no. 5, pp. 103-115, 2019. [Online]. Available: https://aclanthology.org/P19-2008/

[6] Z. Waseem and D. Hovy, "Identifying hate speech targets: Racism, misogyny, and beyond," *Computational Linguistics*, vol. 47, no. 1, pp. 68-85, 2021. [Online]. Available: https://doi.org/10.1162/COLI_a_00368

[7] W. Alorainy, M. Muneer, F. Jahan, and S. Baig, "Dictionary-based hate speech identification in online platforms," *Information Processing & Management*, vol. 57, no. 8, pp. 210-230, 2020. [Online]. Available: https://doi.org/10.1016/j.ipm.2020.102194

[8] P. Fortuna and S. Nunes, "Measuring inter-rater reliability in hate speech classification," *Journal of Data Science*, vol. 16, no. 3, pp. 333-345, 2019. [Online]. Available: https://jds-online.org/article/measuring-inter-rater-reliability-in-hate-speech-classification

[9] A. Georgakopoulou and T. Spilioti, "Digital interactions and toxic speech: Analysis of online communities," *Discourse & Society*, vol. 32, no. 7, pp. 567-582, 2021. [Online]. Available: https://doi.org/10.1177/09579265211012498

[10] B. Vidgen, S. Hale, Z. Margetts, A. Grant, and P. Yasseri, "The challenges of hate speech detection in multilingual settings," *Journal of Artificial Intelligence Research*, vol. 74, no. 2, pp. 89-105, 2022. [Online]. Available: https://doi.org/10.1613/jair.1.12705