Hate Speech Detection on the Twitter Platform

Nirmeen Al-Sheikh 1200200, Raghad Musleh 1200711, Ansam Rihan 1200749

***Abstract*—the project deploys a novel NLP method for com- batting hate speech in social media, an issue that has been seen as the main challenge for online platforms. Our approach includes the utilization of NLP tools that are fast and capable of identifying hate speech stations in any text that matches the Twitter network. Hate speech can be detected most often by using pre-determined data structures (with labels showing whether an instance is hate speech or not). Hate speech is defined here as a direct attack on protected qualities such as race, ethnicity, national origin, disability, religious affiliation, or the use of disparaging language. This technique makes a substantial contribution to creating a safe and inclusive online environment by limiting the detrimental effects of hate speech.**

**The Twitter dataset which is used in the project called The Hate Speech and Abusive (HSAB) is the first ground- breaking Arabic Hate Speech and Abusive Language Dataset of this type. The primary objectives of classification within the dataset are twofold: Binary Classification (Normal, Abusive) and Multi-Class Classification (Normal, Abusive, and Hate Speech).**

**The dataset is divided into training and testing sets, with tweet content and annotations (categorized as Normal, Abusive, or Hate Speech) serving as the features. The Instructions for annotation outlined three label categories as follows:**

**Normal tweets: Instances devoid of offensive, aggressive, insulting, or profane content.**

•

**Abusive tweets: Instances containing offensive, aggressive, insulting, or profane content.**

•

**Hate speech tweets: Instances featuring abusive language directed towards specific individuals or groups and aimed at demeaning or dehumanizing them based on their descriptive identity (such as race, gender, religion, disability, skin color, or belief).**

•

***Keywords: Hate Speech Detection, NLP, Machine Learning, Social Media, Arabic, L-HSAB Dataset, Classification, Data Normalization, Stemming, TF-IDF.***

1. INTRODUCTION

Hate speech over social media increasingly raises concern due to its huge impact on the social lives of people and communities. Hatred and abuse over the internet are very pervasive in the Arab world. For example, 47% of young Arabs were reported to have experienced some form of abuse in a 2020 study. This project uses natural language processing and machine learning to detect hate speech under Arabic content, in order to create an accurately identified and mitigated harmful content project that will make the online environment safer and more inclusive.

1. *Problem statement*

Hate speech is a problem that reflects direct attacks against protected qualities, such as race, ethnicity, and religion. The detection of hate speech from large and diverse social media text data often poses a challenge. This work aims at discussing the development of a robust NLP-based way of classification and detection of hate speech tracking with a machine learning model for efficiency in documentation and maintenance of a safe and more inclusive environment.

1. *Project goals*

* Create an efficient and accurate NLP model for detecting hate speech on social media platforms.
* Use the HSAB dataset to train and test the models, ensuring that they are tailored to Arabic content.
* Implement both binary (Normal, Abusive) and multi-class (Normal, Abusive, Hate Speech) classification to address various levels of abusive language.
* Deploy the optimized model in a real-world setting to detect hate speech more effectively, thereby contributing to a safer and more inclusive online community.
* Improve model performance through techniques such as data normalization and stemming, ensuring robustness and adaptability to changing hate speech trends.
* Create a user-friendly graphical user interface (GUI) with the Tkinter library that allows users to interact with the model, select datasets, add custom hate speech words, and analyze individual tweets.

1. Literature Review

The problem of hate speech detection in social media has gained much attention due to the huge amount of user-generated content. Davidson et al. [1] built up an extensive dataset and a classifier for detecting hate speech on Twitter, proof that automated systems can do so. Their study provided a foundational approach that has influenced many subsequent research projects.

NLP techniques are very vital in hate speech detection. Various studies have been carried out using different NLP techniques to increase the accuracy of the models for such detections. Badjatiya et al. [2] add to the literature using deep learning—specifically, long short-term memory, LSTM, networks and gradient boosting for improved classification performance. Their approach revealed that deep learning can indeed be of help in dealing with the complexities of hate speech.

Applications of most importantly machine learning models span across tasks such as hate speech classification. Waseem and Hovy [3] developed a corpus for the detection of abusive language and classified tweets into machine learning algorithms. Among others, results from their work pointed out the importance of labeled datasets required for the training of effective models. Nobata et al. [4] detected abusive language by the use of machine learning techniques while indicating the usefulness of feature engineering for maximizing model performance.

Detecting hate speech is inherently challenging due to its context-dependent nature and the subtlety of abusive language. Discussion of Schmidt and Wiegand [5] many challenges remain in the definition of hate speech, and variations in language use make it necessary to have models that can adapt to these contexts and languages.

Although most of the research was focused on text in English, there is a fast-growing interest in the detection of hate speech in various languages. For example, Mubarak et al. [6] tackled the problem of detecting abusive language in Arabic, which specifies that various methods developed have to be language-specific. The research in this paper was geared towards the classification of Arabic online hate speech using machine learning and involved an Arabic social media dataset.

Advances in the area of hate speech detection have brought colossal breakthroughs mainly for transformer-based models, especially BERT (Bidirectional Encoder Representations from Transformers). As shown through the work of Mozafari et al. in [7], BERT is a strong model, especially for features capturing contextual nuances that are likely to be omitted in traditional models. Their work underlines the potential of advanced NLP models in enhancing detection accuracy.

1. METHODOLOGY
2. *Data Collection and Preprocessing*
   1. *Data Collection:* File Selection: The user selects CSV files containing training and testing data through a graphical user interface (GUI). These files should contain tweets and their respective class labels (e.g., hate speech, not hate speech).
   2. *Data Preprocessing:*

* Stop words Removal: Arabic stop-words have to be re- moved to clean the text data. Here, the NLTK library will be used, which provides the common stop-words in Arabic.
* Text Cleaning: This is where regular expressions are used to remove URLs, mentions, hash tags, and punctuation from the tweets. This is for sanitizing the text data.
* Custom Hate Speech Words: These are words that users can add to their home language and specify as indicative of hate speech. They are included at pre-processing to tune the model to specific signals indicative of hate speech.

1. *Model Development*
   1. *Feature Extraction:* TF-IDF Factorization: This is the process by which the tweets are cleaned and transformed into numerical features. The process by which this is done is majorly the TF-IDF (Term Frequency-Inverse Document Frequency). This approach helps in establishing the relative importance of the words with respect to the tweets, with a maximum of 5000 features considered to handle the computational complexity for model optimization.
   2. *Model Training:* A logistic regression model will be trained on the TF-IDF features. Logistic regression is chosen for its efficiency and effectiveness in binary classification tasks. The model will be trained by fitting it to the training data, which involves optimizing the weights to correctly classify tweets as either hate speech or not hate speech.
   3. *Model Evaluation:* After training, the model’s performance must be assessed to see if it is capable of accurately classifying tweets. The reserved testing dataset, which was not utilized during training, will be used to test the trained model in order to provide an unbiased evaluation of its performance. The precision, recall, and F1-score of the model are common classification metrics that will be used to assess its efficacy in detecting hate speech, abusive language, and normal tweets. This evaluation clarifies the benefits and drawbacks of the model and serves as a roadmap for any upcoming enhancements or changes.
2. *User Interface*
   1. *GUI Design:* The Tkinter library will be utilized to create a user-friendly graphical user interface. The GUI includes functionalities for selecting training and testing files, running the model, adding custom hate words, and checking individual tweets.
   2. *Interactivity:*

* File Selection: User can easily select the CSV file for training and testing through GUI file dialog windows.
* Run Model: A GUI button allows running the training and then evaluation of the model. This enables a straightforward training of the model on user-provided data with immediate feedback about the performance of the model.
* Add Custom Words: The user can input custom hate speech words via the GUI. These are then fed into the model for analysis, allowing some tailoring to suit certain needs or emerging trends in hate speech.e.
* Check Tweet: The GUI allows the user to introduce single tweets that will be analyzed with the previously trained model. Results, including class and hate words, are displayed in the interface.

1. OUR PROPOSED WORK & RESULTS
2. *For the Learning Model*

The training dataset contains an imbalance among the classes as shown in Figure 1:

* Normal Tweets: Approximately 3000 instances.
* Abusive Tweets: Approximately 1300 instances.
* Hate Tweets: Less than 300 instances.



Figure 1 Training Set Class Distribution.

The test dataset has a similar class distribution shown in Figure 2:

* Normal Tweets: Over 700 instances.
* Abusive Tweets: Around 300 instances.
* Hate Tweets: Less than 100 instances.

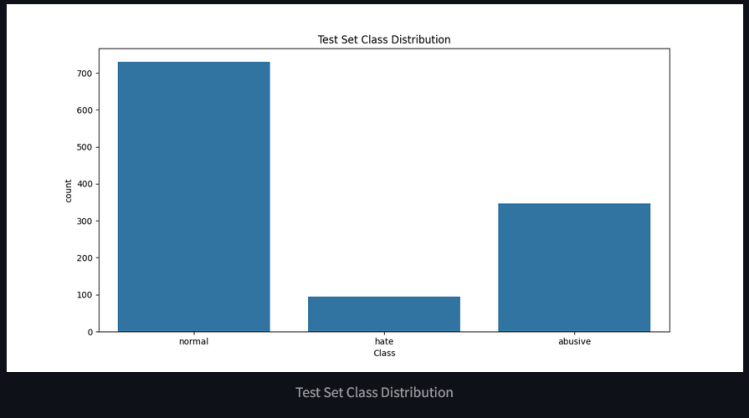


Figure 2 Testing Set Class Distribution

The trained logistic regression model was evaluated on the test dataset. These results we got before the stemming and normalization to our data, the accuracy and performance of the model is as shown in Figure 3:

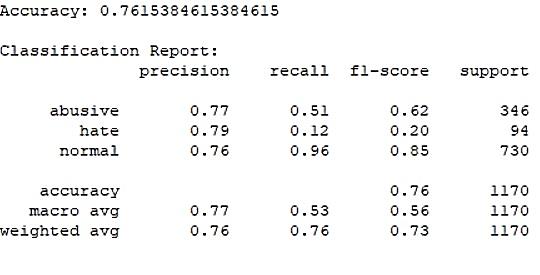


Figure 3 Accuracy& Classification Report before Data Normalization & Stemming

The classification report and accuracy metrics provide a detailed evaluation of our model's performance in detecting abusive, hate, and normal tweets. The model achieves an overall accuracy of 76.15%, indicating it correctly classifies 76.15% of the tweets in the test set.

For abusive tweets, the model demonstrates a precision of 0.77, meaning 77% of the tweets classified as abusive are correctly identified. However, the recall is lower at 0.51, indicating that the model only detects 51% of all actual abusive tweets. This results in an F1-score of 0.62, reflecting a moderate balance between precision and recall.

The performance on hate tweets is notably poor. While the precision is relatively high at 0.79, the recall is extremely low at 0.12, meaning the model identifies only 12% of all actual hate tweets. This low recall leads to a low F1-score of 0.20, highlighting the model's difficulty in effectively detecting hate speech.

For normal tweets, the model performs well, with a precision of 0.76 and a high recall of 0.96, indicating that it correctly identifies 96% of all normal tweets. The resulting F1-score is 0.85, demonstrating robust performance for this class.

**We enhanced the model by stemming and normalization the data, the accuracy and performance of the model is as shown in Figure 4:**

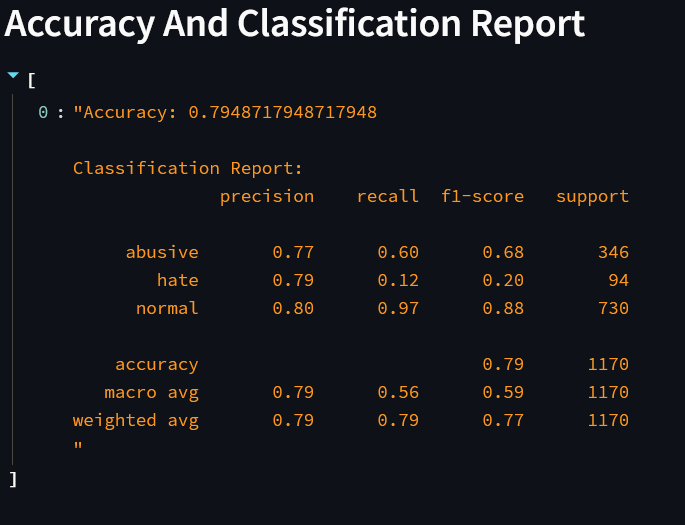


Figure 4 Accuracy & Classification Report After Data Normalization and Stemming

The model has an accuracy of 79.49%, hence correctly predicting almost 79.49% of the instances.

For the abusive class, the precision comes to 0.77, meaning that 77% of the cases that were predicted by the model as abusive were abusive. However, the recall is just 0.60, which means that the model spotted only 60% of the actual abusive instances. The F1-score is 0.68, hence balancing the precision and recall, and therefore showing the model's moderate performance in identifying abusive content.

This model is highly precise for the hate class, at 0.79, so when it predicts an instance as hate, it truly is 79% of the time. In contrast, recall is very low, at 0.12, saying that a large number of instances that truly are instances of hate were missed. Correspondingly, the F1 score was poor for this class at 0.20, showing bad overall performance in this category.

The normal class does very well with a high precision of 0.80 and an extremely good recall of 0.97. This means it is excellent at picking up normal instances—it gets 97% of them. The F1-score here again is very robust, at 0.88, for the classification of normal content.

The overall accuracy of the model is 0.79, which indicates how well it classifies all instances. According to the macro average, giving equal weight to all classes, the precision for this model comes in at 0.79, recall at 0.56, and an F1-score of 0.59. From these metrics, one could infer that, while this model does very well in terms of making correct predictions for most instances, it actually performs relatively poorly with regard to recall for all classes. The weighted average gives quite similar precision and recall to the overall model performance metrics.

Table 1 Comparison of Model Performance before and after Data Normalization and Stemming

|  |  |  |
| --- | --- | --- |
| **Metric** | **Initial Model** | **Enhanced Model** |
| **Accuracy** | 76.15% | 79.49% |
| **Abusive Tweets** |  |  |
| Precision | 0.77 | 0.77 |
| Recall | 0.51 | 0.6 |
| F1-Score | 0.62 | 0.68 |
| **Hate Tweets** |  |  |
| Precision | 0.79 | 0.79 |
| Recall | 0.12 | 0.12 |
| F1-Score | 0.2 | 0.2 |
| **Normal Tweets** |  |  |
| Precision | 0.76 | 0.8 |
| Recall | 0.96 | 0.97 |
| F1-Score | 0.85 | 0.88 |

The table demonstrates that the upgraded model, which includes data normalization and stemming, increases overall accuracy from 76.15% to 79.49%. It is more effective at detecting abusive tweets, with a recall increase of 0.51 to 0.60 and an F1-score improvement of 0.62 to 0.68. Precision and recall have increased for typical tweets, resulting in a higher F1-score of 0.88 against 0.85, respectively. However, the performance of hate tweets remains unchanged, with low recall and F1-score, indicating that more improvements are required.

1. *Keyword-Based Arabic Hate Speech Detection*

Using a keyword-based method, our Arabic hate speech recognition software determines if a tweet is hate speech, abusive, or normal. It uses a Tkinter-based GUI to allow users to load files containing predefined abusive and hate speech phrases, create custom hate speech words, and load tweets for classification. Tweets are preprocessed to remove URLs, mentions, punctuation, and stopwords before being stemmed with the ISRIStemmer. The classification is carried out by identifying the presence of these words in the tweets. Although this method is quite accurate in identifying predefined keywords, it does not use machine learning techniques to understand patterns or improve over time, which limits its capacity to handle nuanced or previously overlooked content.

1. *User Interface*

The user can easily select the CSV file for training and testing through the GUI as shown in Figure 5:

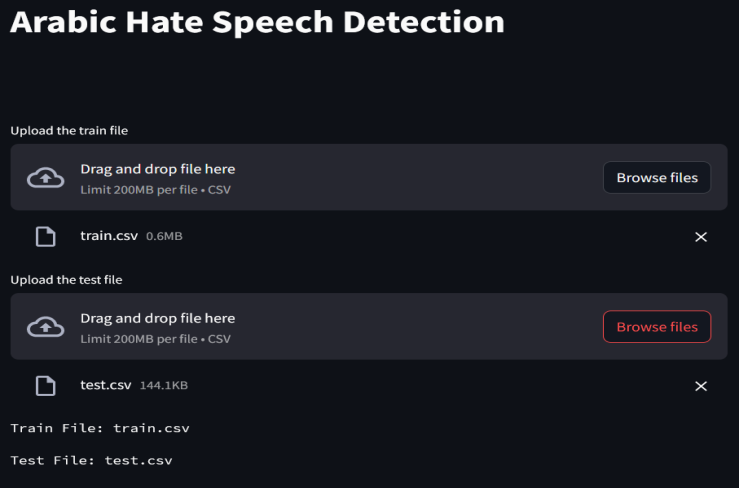


Figure 5 Select Training & Testing Data Files

The user can input custom hate speech words via the GUI as shown in the Figure 6:

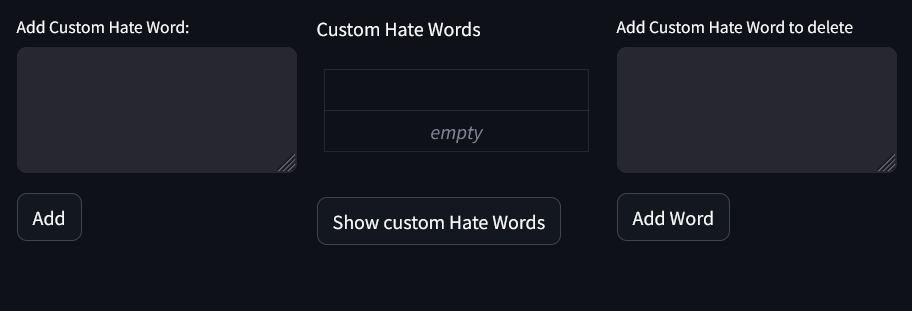


Figure 6 Customizing Hate Words

The following pictures are some tweet sample cases and tests to our model:



Figure 7 Sample Tweets

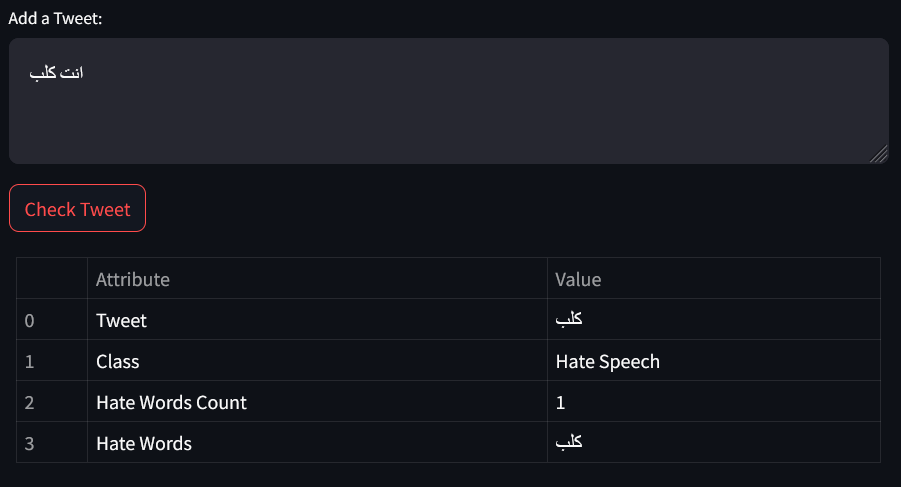
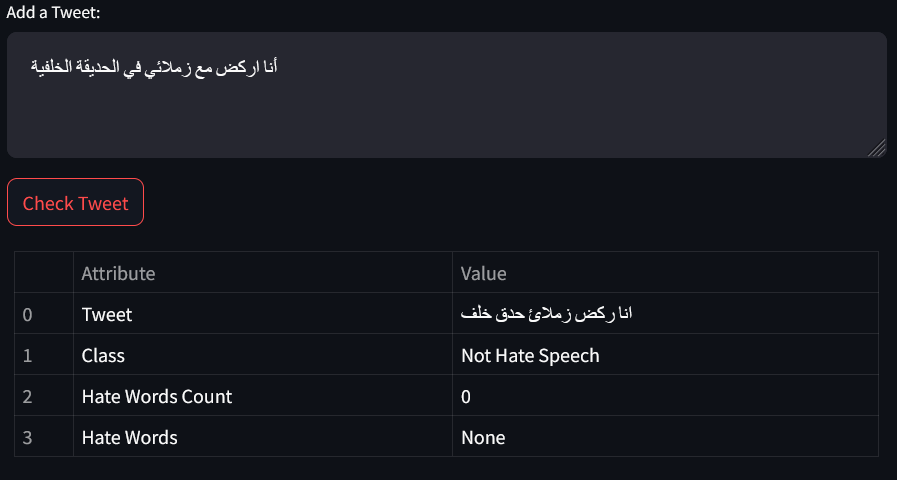


Figure 8 Check Tweet Samples I/O

(a)



(b)

1. CONCLUSION

In conclusion, our study aimed to create and optimize a natural language processing model for identifying hate speech in Arabic social media posts. We used the L-HSAB dataset and approaches such as data standardization and stemming to increase the model's accuracy and applicability.

We used binary and multi-class classification to categorize tweets as normal, abusive, or hate speech. The results show promising improvements in accuracy after preprocessing, particularly in identifying abusive and normal tweets. However, because hate speech is nuanced and context-dependent, effective detection remains difficult. Moving forward, further refinement of the model and research into advanced NLP techniques will be critical for mitigating the negative effects of hate speech online and promoting a safer digital environment for all users.

***References***

1. T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated Hate Speech Detection and the Problem of Offensive Language," in Proceedings of the 11th International AAAI Conference on Web and Social Media (ICWSM), 2017.
2. P. Badjatiya, S. Gupta, M. Gupta, and V. Varma, "Deep Learning for Hate Speech Detection in Tweets," in Proceedings of the 26th International Conference on World Wide Web Companion, 2017.
3. Z. Waseem and D. Hovy, "Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter," in Proceedings of the NAACL Student Research Workshop, 2016.
4. C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang, "Abusive Language Detection in Online User Content," in Proceedings of the 25th International Conference on World Wide Web, 2016.
5. A. Schmidt and M. Wiegand, "A Survey on Hate Speech Detection using Natural Language Processing," in Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media, 2017.
6. H. Mubarak, K. Darwish, and W. Magdy, "Abusive Language Detection on Arabic Social Media," in Proceedings of the First Workshop on Abusive Language Online, 2017.
7. M. Mozafari, R. Farahbakhsh, and N. Crespi, "A BERT-based Transfer Learning Approach for Hate Speech Detection in Online Social Media," in Proceedings of the European Conference on Artificial Intelligence (ECAI), 2020.
8. H. Watanabe, M. Bouazizi, and T. Ohtsuki, "Hate Speech on Twitter: A Pragmatic Approach to Collect Hateful and Offensive Expressions and Perform Hate Speech Detection," IEEE Access, vol. 6, pp. 13825-13835, 2018.
9. Z. Zhang, D. Robinson, and J. Tepper, "Detecting Hate Speech on Twitter Using a Convolution-GRU Based Deep Neural Network," in Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL), 2018.