NLP Based Approach for Detection of Hate and Offensive Speech on Social Networks

Aniruddho Chatterjee (caswapan@iu.edu)[1],Piyush Chaudhari (piyrchau@iu.edu)[1],Shivani Pal (shipal@iu.edu)[1],Dheeraj Manchandia (dmancha@iu.edu)[1]

Abstract
In today's digital age, social media platforms have become a common avenue for communication, but unfortunately, they have also become a breeding ground for hate speech and offensive language. The prevailing methods of detection of hate speech like lexical detection tend to have lower precision in detecting hate speech since they rely on a few particular keywords and phrases flagged as hate speech. Such traditional methods of detecting hate speech can be effective but often miss out on subtle nuances and implicit forms of hate speech. Therefore, we have proposed an NLP-based approach using Robustly Optimized BERT Approach (RoBERTa) that can detect not only explicit and direct forms of hate speech but also subtle and implicit forms. By taking into account the broader context in which the language is used, our model can better differentiate between language that is intended to be harmful and language that is not.

Keywords
Hate Speech Detection, Natural Language Processing, LSTM, Distil BERT, RoBERTa

[1] *Luddy School of Informatics, Computing, and Engineering, Indiana University, Bloomington, IN, USA*

Contents

1. Problem and Data Description

The problem we aim to solve is the detection of offensive language in given text data.

Hate speech or offensive language are prevalent and growing in online platforms. These lead to severe consequences such as psychological trauma, social exclusion, and even violence. Detecting such language can help in preventing harassment and promote a safe and inclusive online environment.

By analyzing online text, and data mining algorithms, we aim to identify patterns and features that differentiate between hate/offensive speech and non-hate speech to enable platforms to moderate content effectively and reduce the harm caused by hate speech also ensuring the safety and well-being of individuals using these platforms.

We aggregated data from the following sources:

1. Automated Hate Speech Detection

   Davidson, T., Warmsley, D., Macy, M., and Weber, I. (2017). Automated Hate Speech Detection and the Problem of Offensive Language. Proceedings of the International AAAI Conference on Web and Social Media, 11(1), 512-515. [Research Paper Link] [Dataset Link]

2. Cyberbullying Data [Dataset link]
   The cyberbullying dataset is curated by automatic generation of balanced data by using a semi-supervised online Dynamic Query Expansion (DQE) process by J. Wang, K. Fu, C.T. Lu in their paper [Link to Paper]. Overall, it has 37218 cyberbullying tweets and 7682 non-cyberbullying tweets.

3. Twitter Sentiment Analysis

   The Twitter sentiment analysis dataset is generated from tweets on Twitter and flagged as hate or offensive based on sexist and racist comments. [Dataset Link]

4. HatespeechData [Dataset Link]:
   The text data (abusive and non-abusive) is collected from three conversational AI systems gathered 'in the wild': an opendomain social bot, a rule-based chatbot, and a task-based system. Dataset has 10068 number of abusive records and 2029 non-abusive records. [Link to Paper]

2. Data Preprocessing & Exploratory Data Analysis

2.1 Handling Missing Values

Since there were only a few missing values, the related records were excluded and removed.

2.2 Exploratory Data Analysis

Following are the steps undertaken in EDA process:

i. Data Cleaning

   a. Converting text into lower case

   b. Removing emojis

   c. Removing non-alphabet characters (punctuation marks, line breaks, bogus, etc.)

   d. Removing user Ids and hashtags from Twitter data.

   e. Removing URLs

   f. Removing multiple spaces and characters where they are followed by one or more instances of the same character

ii. Handling contractions: Contractions are created by shortening words and replacing the omitted letters with apostrophes. It is important to do this before the removal of punctuation. Failure to do so may negatively impact the accuracy and efficiency of NLP tasks, leading to erroneous results or incomplete information extraction. We handled it by creating a contraction dictionary and replacing the contractions in the test with their un-shortened versions.

iii. Corpus Statistics: In our combined dataset, we have a total of 50000 offensive and 60000 non-offensive texts.

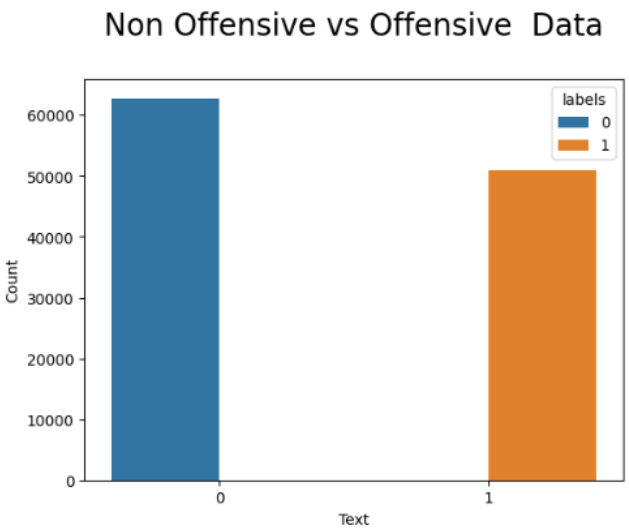

Figure 1. Offensive Vs Non-Offensive Count

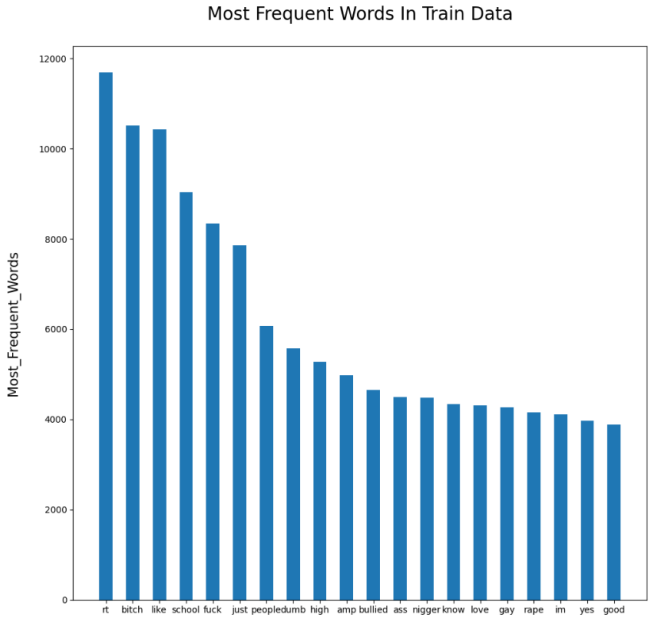

Figure 2. Most Frequent Words

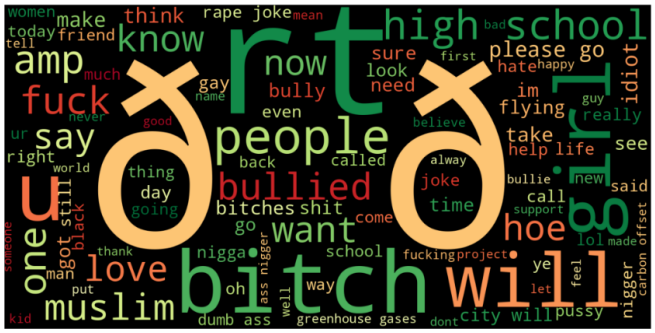

Figure 3. Word Cloud

3. Algorithm and Methodology

We know that transformers were introduced because of their efficiency in text detection where they are able to learn long-range dependencies between text tokens. This is important for text detection because text can be arranged in a variety of ways, and transformers are able to learn the relationships between text tokens regardless of their order. Additionally, transformers are able to learn from large amounts of data, which is important for text detection because there is a lot of variation in the way that text can be presented. In this project, we have considered three models Long Short-Term Memory Network (LSTM), DistilBERT, and Robustly Optimized BERT Approach (RoBERTa).

1. **LSTM**:
   Recurrent neural networks (RNNs) of the LSTM variety are useful for learning long-term dependencies in

sequential data. RNNs are a particular kind of neural network that can handle sequential data, like text or audio. However, RNNs can suffer from the vanishing gradient problem, which makes it difficult for them to learn long-term dependencies. LSTMs were created to solve this issue.

LSTMs employ a gating mechanism to control the information flow throughout the network. The gating system is composed of the input gate, forget gate and output gate. How much input current can pass through the network is controlled by the input gate. The forget gate determines how much of the previous state is forgotten. How much of the network's current state is sent out is controlled by the output gate.

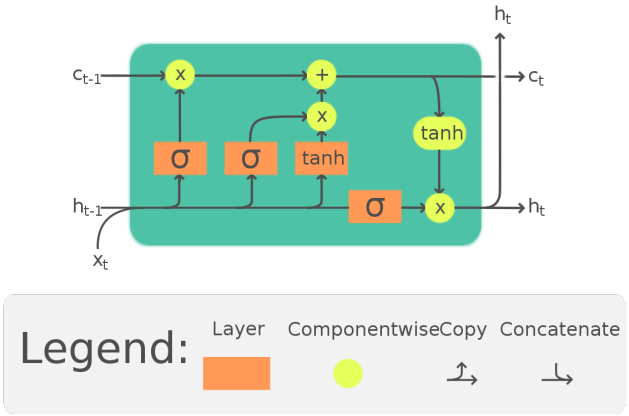

Figure 4. LSTM Cell Architecture

2. **BERT**:
Bidirectional Encoder Representations from Transformers, or BERT, is a language model that was released in 2018 by Google AI researchers. A vast dataset of text and code was used to train the neural network known as BERT.
To solve the shortcomings of earlier language models, BERT was developed. Previous language models could only learn the meaning of words and phrases in the context of the words that came before them since they were trained on material that was only unidirectionally labeled. Contrarily, BERT is trained on bi-directionally labeled text, allowing it to understand the meaning of words and phrases in the context of the words that come before and after them. Due to its increased strength over earlier language models, BERT has been demonstrated to be cutting-edge in a number of natural language processing tasks.

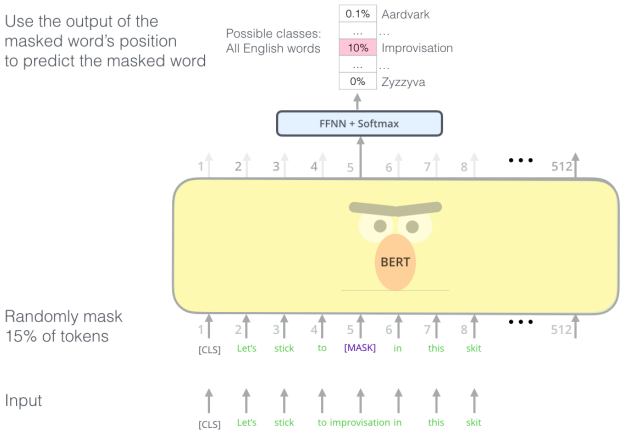

Figure 5. BERT Architecture

(a) *Distil-BERT*: Researchers at Google AI created DistilBERT, a more compact and effective version of BERT, in 2019. Using a distillation technique, DistilBERT is trained to imitate the actions of a bigger, more intricate model, such as BERT. On a range of natural language processing tasks, DistilBERT outperforms BERT despite being considerably smaller and quicker to train.
To remedy the shortcomings of BERT, DistilBERT was developed. BERT is a big, complicated model that can be expensive to train and use computationally. DistilBERT is a more compact, effective model that is simpler to train and use. As a result, DistilBERT is a more affordable choice for academics and developers who want to benefit from BERT power without incurring additional expenses.

(b) *RoBERTa*: Researchers at Facebook AI created the language model known as RoBERTa, or Robustly Optimized BERT Pretraining approach, in 2019. A modified variant of BERT called RoBERTa has been proven to produce cutting-edge outcomes on a number of natural language processing tasks. Although RoBERTa is a strong language model, it does have several drawbacks. One drawback is that, for some tasks, it could not be as accurate as BERT. Another drawback is that RoBERTa might not be as resistant to extremely lengthy or extremely short text sequences as BERT.

4. Experiments and Results

1. Accuracy Results The accuracy results show that LSTM has an accuracy of 86.36 % on the Validation set and 89.93 % on the testing set. Distil BERT model performs much better than the LSTM model with the validation set accuracy of 91.02 % and test set accuracy of 90.04

%. RoBERTa model performs quite similarly to that of Distil BERT with the validation set accuracy of 92.45 % and test set accuracy of **90.70** %.

	Model	Accuracy on Validation Set	Accuracy on Test Set
1	LSTM	86.36%	89.93%
2	DestilBERT	91.02%	90.04%
3	RoBERTa	92.45%	90.70%

Figure 6. Accuracy Results

2. Model Validation and Evaluation metrics
   Below are the two metrics used to evaluate and validate

   (a) ROC Curve:
   ROC (Receiver Operating Characteristic) curve is a measure of the performance of a binary classification model. The ROC curve is a graphical representation of the trade-off between the true positive rate (TPR) and the false positive rate (FPR) at various classification thresholds.

   (b) Confusion Matrix
   A confusion matrix is a table that summarizes the performance of a machine learning model on a binary classification problem. It compares the predicted values of the model to the actual values, breaking down the number of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) in a tabular format.

3. LSTM Model
   Below is the ROC curve of the LSTM model

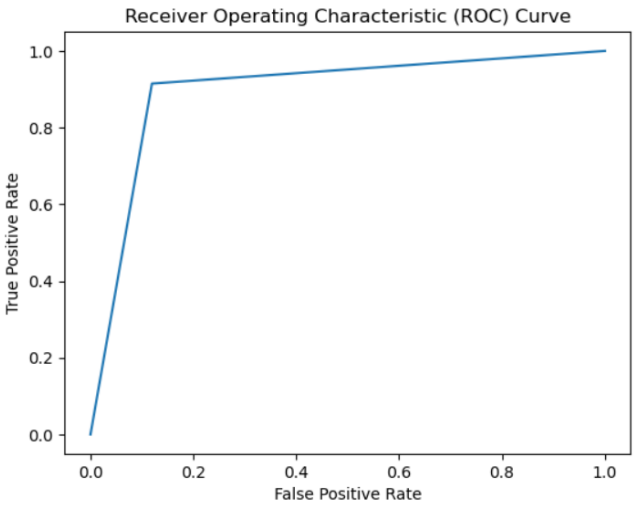

Figure 7. LSTM ROC Curve

The confusion matrix of the LSTM model shows that 13490 records were perfectly classified as offensive and 17192 records were classified as non-offensive. 1602 records were originally offensive but were classified as non-offensive. Whereas, 1830 records were originally non-offensive but were wrongly classified as offensive text.

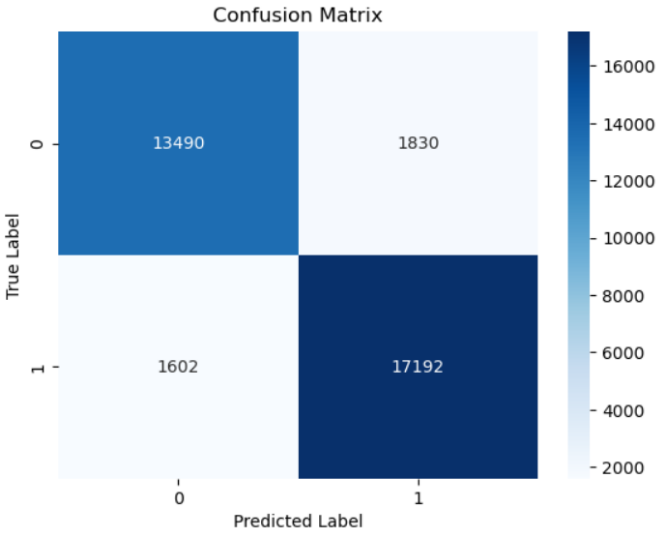

Figure 8. LSTM Confusion Matrix

4. Distil BERT Model
   The ROC curve of Distil BERT model performs better than that of the LSTM model. The curve achieves the peak quickly and the Area Under Curve (AUC) is maximum. This shows that the Distil BERT model performs better than the LSTM model.

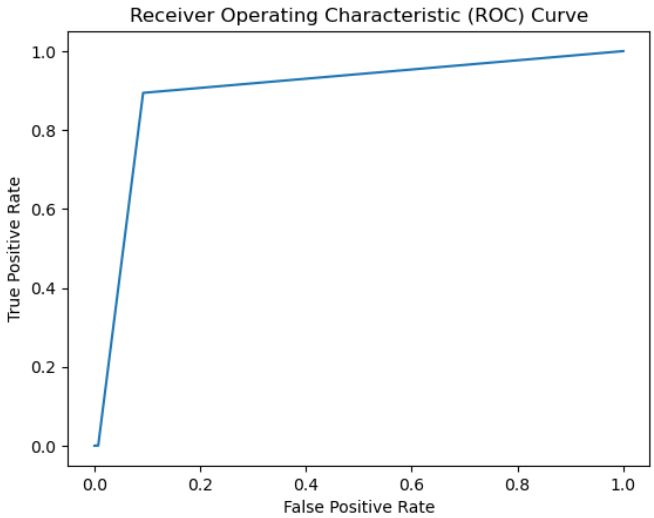

Figure 9. Distil BERT ROC Curve

The confusion matrix also shows similar trends with 16806 records rightly classified as Offensive and 14019 as non-offensive. There is a reduction in the number of records that were originally non-offensive but were classified as offensive.

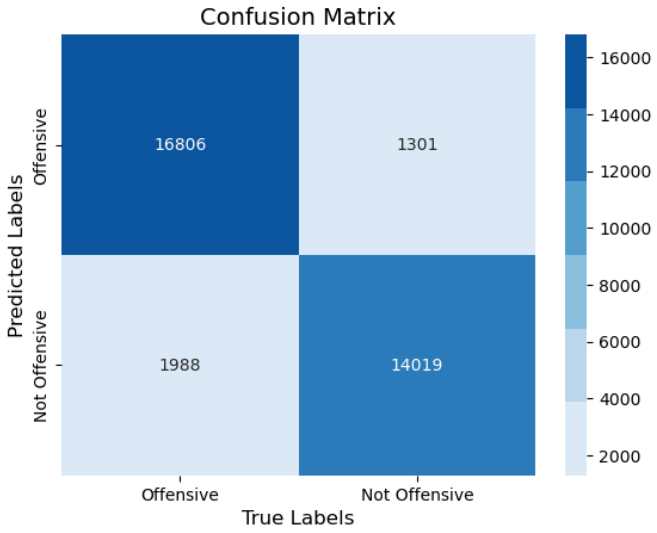

Figure 10. Distil BERT Confusion Matrix

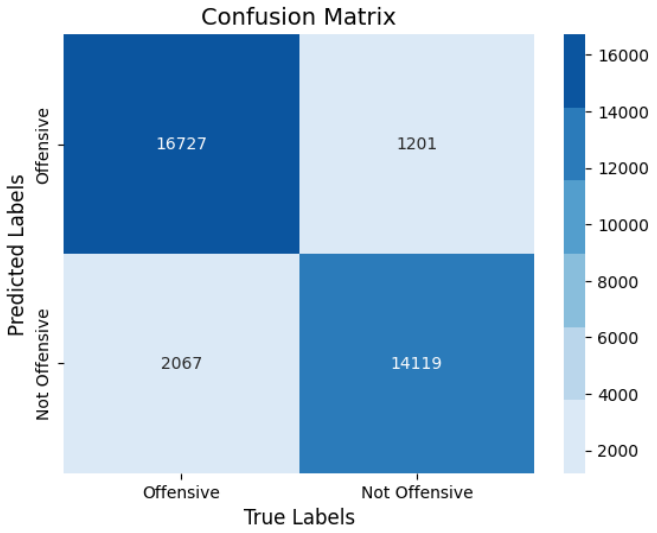

Figure 12. RoBERTa Confusion Matrix

The RoBERTa ROC Curve is very similar to that of Distil BERT since both models are quite similar.

5. RoBERTa Model

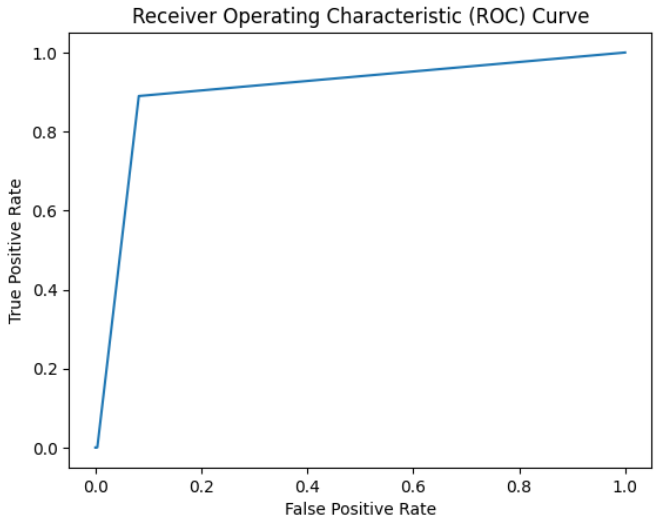

Figure 11. RoBERTa ROC Curve

The RoBERTa Confusion Matrix is also very similar to that of Distil BERT since both models are quite similar. The only difference is that there is a slight reduction in the number of records originally non-offensive but classified as offensive by the model.

5. Deployment and Maintenance

1. Backend:
   Tech Stack: Python, Flask
2. Frontend:
   Tech Stack: React JS

The landing page of our website is as below.
Here, you can navigate to our text checker page
There are links to the About Us, Our Product, and Contact Us pages.

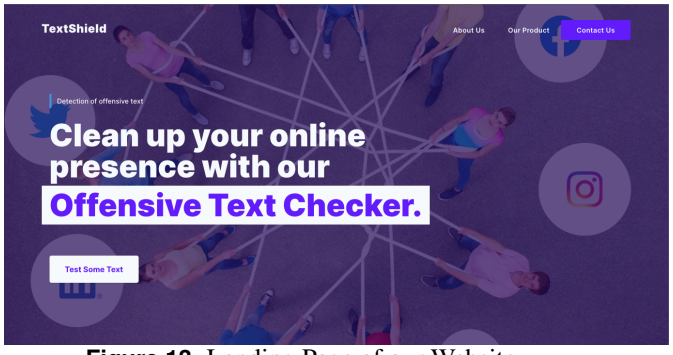

Figure 13. Landing Page of our Website

Below you can see the screenshot of our Text Checker page.
Here you can enter any text in the text box on the left side of the page.
On clicking the Test button, our algorithm will take the text and determine whether it is offensive or not.

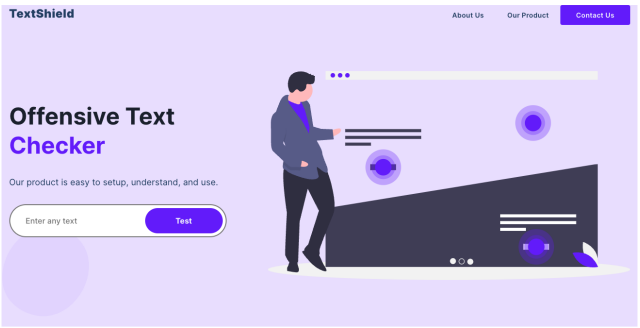

Figure 14. Text Checker Page

Below is a test case of offensive text:
If we enter the text: How dumb can you be?
Our algorithm will classify this text as Offensive and display this on the screen

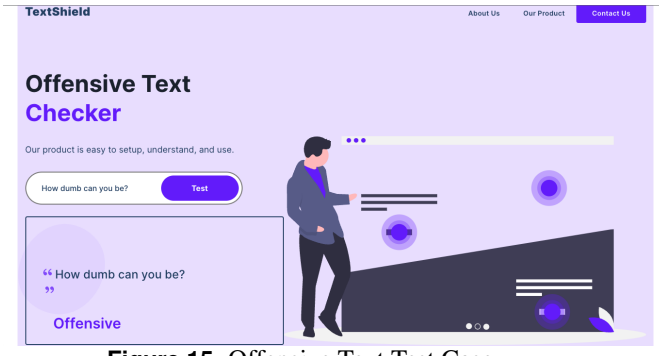

Figure 15. Offensive Text Test Case

Below is a test case of non-offensive text:
If we enter the text: Dumb has 4 letters
Our algorithm will classify this text as Non-Offensive. The traditional method upon encountering the word dumb would automatically categorize this as offensive, but since our algorithm takes context into consideration, it successfully classifies this as non-offensive in the second sentence.

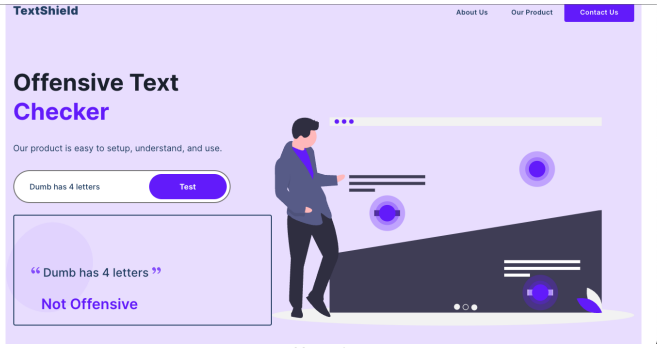

Figure 16. Non-Offensive Text Test Case

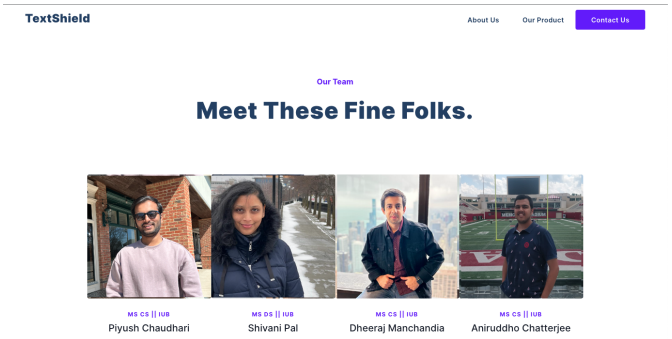

Figure 17. Our Team

6. Summary and Conclusions

In this study, we developed a high-quality text corpus of English hate and offensive speech by aggregating four datasets. To the best of our knowledge, the merged dataset has AI bot - Human interaction text, twitter tweets and other hate/non-offensive texts. We conducted an extensive empirical analysis by taking into consideration three models LSTM, DistilBERT and RoBERTa. From results, it is evident that word embeddings extracted by RoBERTa and DistilBERT are best and among both RoBERTa is outperforming. Finally, there are several future directions for our work. Fine tuning best performing model with large dataset, along with different set of hyperparameters.

Acknowledgments

We would like to express our sincere gratitude to the following individuals and organizations who contributed to this research project:

First and foremost we would like to thank Indiana University, Bloomington for providing the infrastructure resources such as supercomputers to execute the machine learning models.

We would also like to thank Dr. Hasan Kurban and Associate Instructors for their guidance and support throughout the research process. Their insights and feedback were invaluable in shaping the direction of this study.

Finally, we would like to express our appreciation to our families and friends who provided emotional support and encouragement throughout the research process.

Any errors or omissions in this paper are solely the responsibility of the authors.

References

[1] *Hind Saleh and Areej Alhothali and Kawthar Moria, Detection of Hate Speech using BERT and Hate Speech Word Embedding with Deep Model, Applied Artificial Intelligence,2023. https://arxiv.org/ftp/arxiv/papers/2111/2111.01515.pdf*

[2] *https://huggingface.co/docs/transformers/v4.28.1/en/tasks/sequence_classification*

[3] *https://ai.facebook.com/blog/ai-advances-to-better-detect-hate-speech/*

[4] *Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. ArXiv. /abs/1907.11692*

[5] *Alammar, J (2018). The Illustrated Transformer [Blog post]. Retrieved from https://jalammar.github.io/illustrated-transformer/*

[6] *Alammar, J (2018). The Illustrated Transformer [Blog post]. Retrieved from https://jalammar.github.io/illustrated-transformer/*