



Deep Learning for Toxic Comment Detection

Final Project (Applied Mathematics Concepts For Deep Learning)

By-

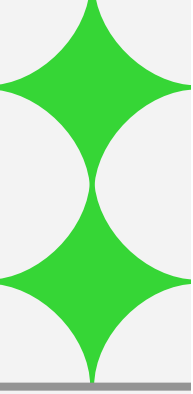
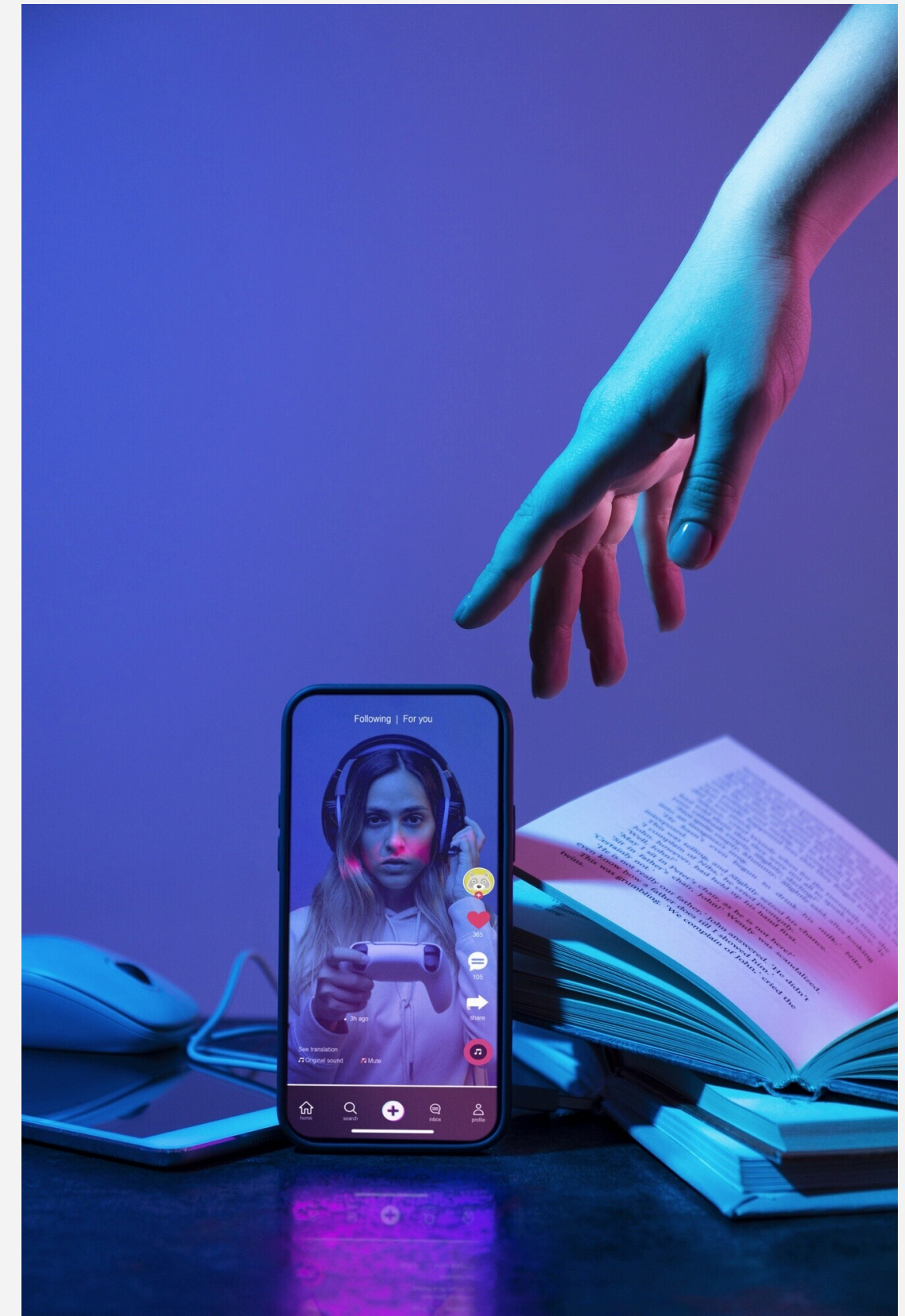
Pranali Karande(101471932)

Isha Jayswal(101510506)



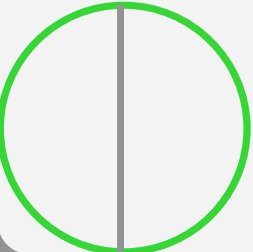

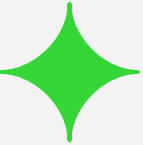
Introduction:

- This project focuses on developing a deep learning model to automatically detect and classify toxic comments in online discussions.
- The proliferation of toxic behavior, including hate speech, insults, and threats, poses significant challenges to maintaining respectful and safe online environments.
- The goal of this project is to leverage natural language processing (NLP) techniques and deep learning algorithms to mitigate these challenges by accurately identifying and categorizing toxic comments.





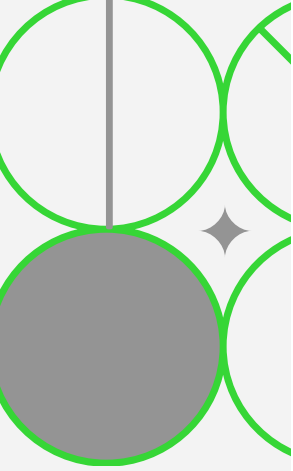
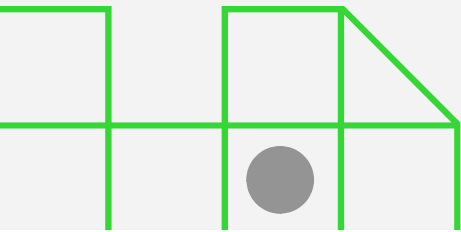
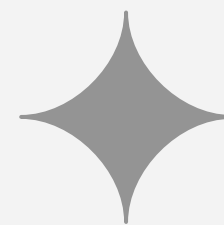
Dataset Overview:

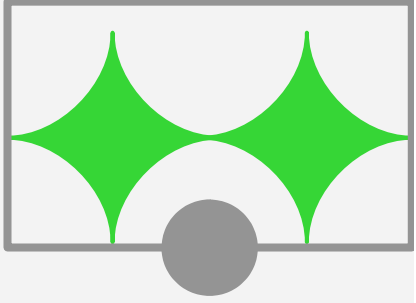
- The dataset used in this project is sourced from Kaggle and contains labeled comments from wikipedia
 - Link: <https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/data>
 - The dataset comprises a large number of comments, with toxicity labels across categories such as toxicity, severe toxicity, obscenity, threat, insult, and identity hate.
 - The dataset undergoes preprocessing steps like cleaning and tokenization to prepare it for model training.
- 
- 
- 

Model Architecture:

- The model architecture comprises a Sequential neural network with layers including an embedding layer, bidirectional LSTM layer, and multiple dense layers for classification.
- Each component plays a crucial role in transforming text data into numerical vectors, capturing sequential dependencies, and making predictions.

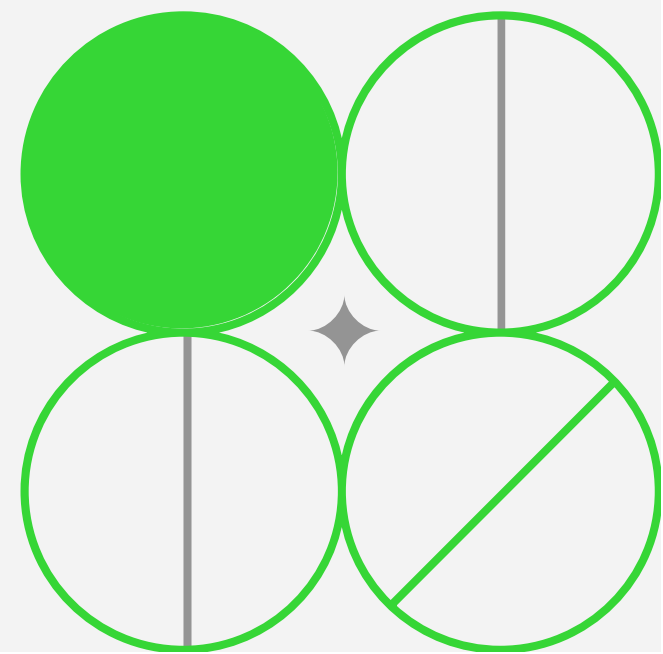
```
model=Sequential()  
#Create the embedding layer  
model.add(Embedding(MAX_FEATURES+1,32))  
#Bidirectional LSTM Layer  
model.add(Bidirectional(LSTM(32,activation='tanh')))  
#Feature extractor fully connected layers  
model.add(Dense(128,activation='relu'))  
model.add(Dense(256,activation='relu'))  
model.add(Dense(128,activation='relu'))  
#Final layer  
model.add(Dense(6,activation='sigmoid'))
```





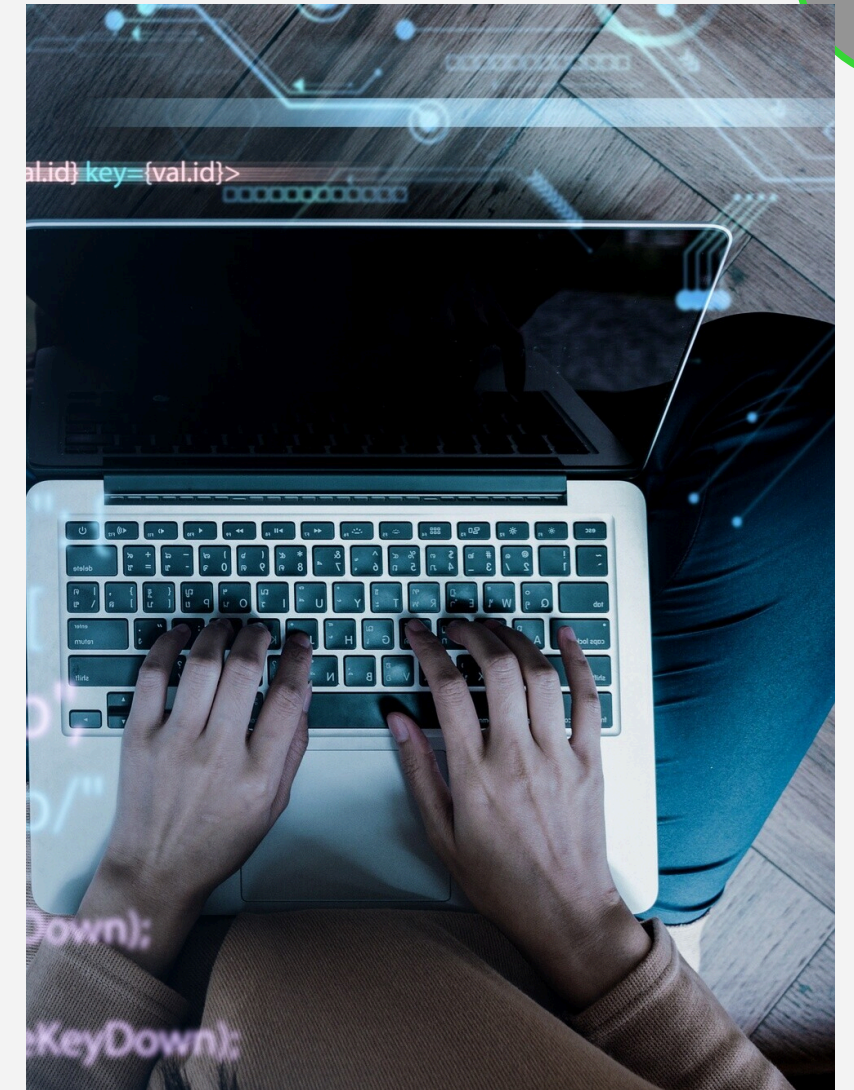
Training Process:

- The model is trained using an optimization algorithm over a specified number of epochs.
- During training, the model achieves a loss of 0.0622 and a validation loss of 0.0457, indicating its learning progress.
- Insights into the model's convergence behavior and optimization strategies employed during training provide valuable information about its performance.



Model Evaluation:

- Evaluation metrics: Precision, recall, and accuracy metrics are used to assess the model's performance in correctly classifying toxic comments.
- Performance summary: The model achieves high precision (0.8207), recall (0.6717), and accuracy (0.9945) scores across various toxicity categories.
- Visualizations: Visualizations of evaluation metrics, if available, provide additional insights into the model's performance.



```
In [70]: print(f'Precision: {pre.result().numpy()}, Recall: {re.result().numpy()}, Accuracy: {acc.result().numpy()}')
```

```
Precision: 0.8207114934921265, Recall: 0.6716632843017578, Accuracy: 0.994546115398407
```

Analysis of predictions:

- An example of comments classified by the model along with their predicted results for each toxicity category provide insights into the model's decision-making process:

```
In [122]: score_comment('I hate you')  
1/1 [=====] - 0s 50ms/step  
Out[122]: 'toxic: True \n severe_toxic: False \n obscene: False \n threat: False \n insult: False'
```

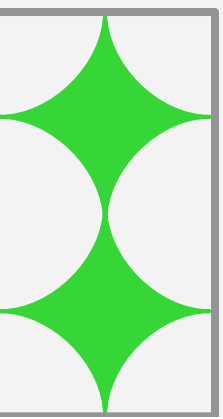
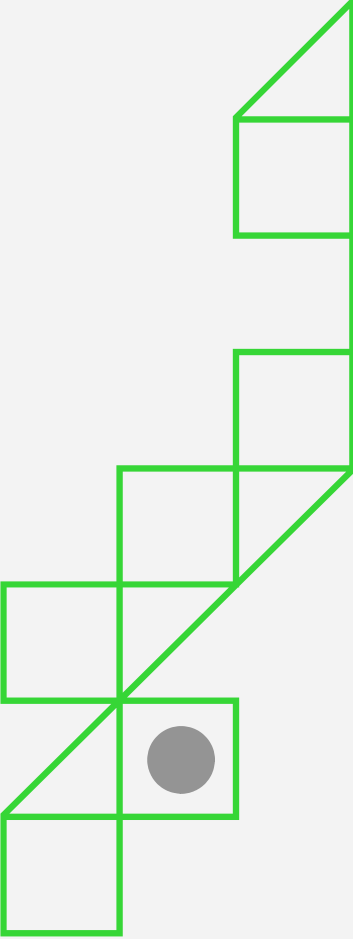
- Understanding the model's decision-making process enhances interpretability and reliability.





Integration with GradioOverview of integration:

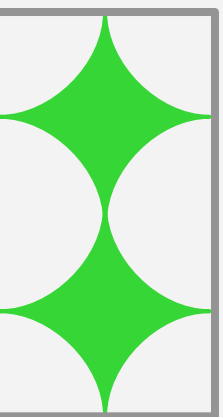
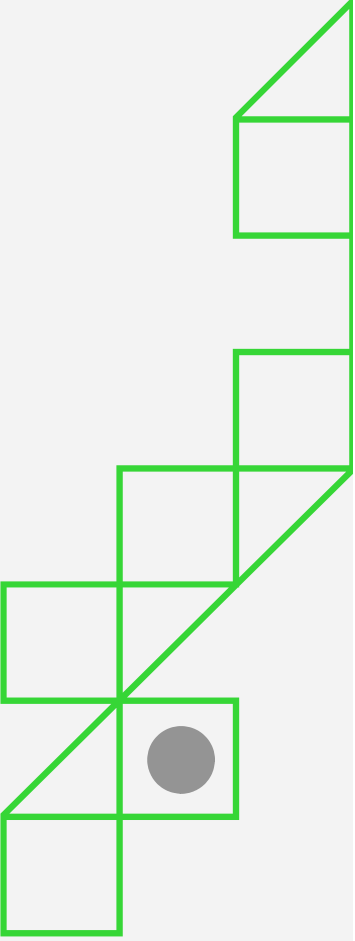
- The model is integrated into a user-friendly interface using Gradio, allowing users to interact with the model and receive toxicity predictions in real-time.
- Benefits of Gradio: The benefits of Gradio, such as ease of use and real-time predictions, enhance accessibility and usability for end-users.

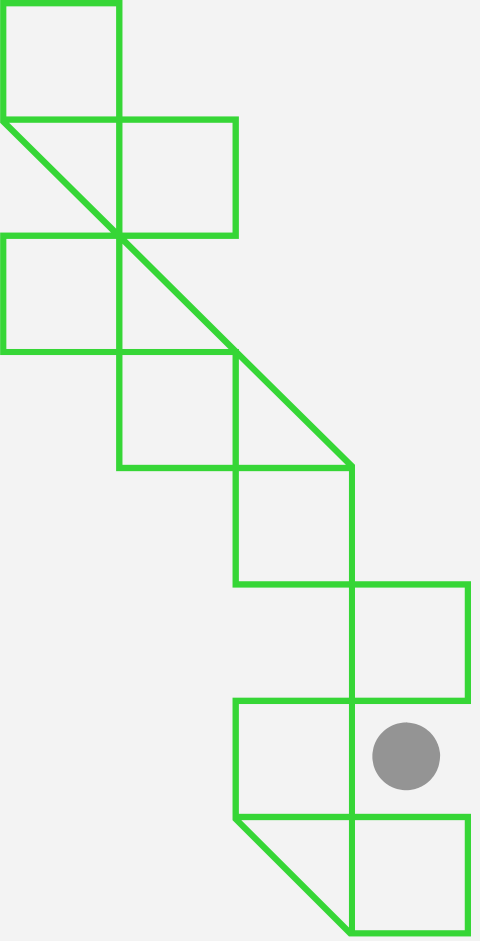




Conclusion:

- The project demonstrates the effectiveness of deep learning techniques in addressing social challenges such as toxic behavior in online discussions.
- By accurately identifying toxic comments, the developed model contributes to fostering healthier and more respectful online communities.
- Future research efforts can focus on refining the model architecture, increasing training data diversity, and exploring additional features to enhance performance and usability further.





Thanks!

