Progress report 2

Team Members:

Aysha Mohamed
Mostafa Hasona
Youssef Muhammed Abd El-alim

Introduction:

Our project aims to detect offensive language and hate speech in Arabic social media posts using emojis as extralinguistic anchors. By focusing on emojis, we aim to circumvent the limitations of keyword-based methods. The research leverages various machine learning models, including BERT and fast Text, to enhance detection accuracy and robustness and make it more advance that we had web application

Objectives:

1. Pre-process Arabic text data.
2. Utilize pre-trained models like BERT and fast Text for classification.
3. Evaluate model performance.
4. Explore adversarial robustness by perturbing input data.

Progress Overview:

1. Data Collection and Preparation

- Collected and loaded tweet data and corresponding hate speech labels from specified URLs.
- Renamed columns for consistency and clarity.
- Preprocessed tweets by normalizing text, tokenizing, removing stop words, and cleaning HTML and non-Arabic content.

- Further data augmentation by altering input tweets to create new examples and test model robustness.

2. Model Setup and Loading

- Loaded BERT-based Qarib model and tokenizer using the Transformers library.

- Initialized the model with pre-trained weights.
- Loaded the fastText model for word embeddings.

- Training and fine-tuning models for optimal performance.

3. Adversarial Attack Method

Developed a script for perturbing tweets by replacing words with synonyms or similar terms using a black-box greedy approach with three stages:

- Replacing a word with the mask (Token)
- Applying Bert to find substitute words
- Utilizing a pretrained word vector model to find the most similar substitute to word

(1) Replace a word with the (mask) token
(2) Unmask using a Bert based model
(3) Find Most similar candidate

4. Evaluation and Results

- Classified cleaned tweets using the model and calculated accuracy for the hate speech class.
- Analyzed the impact of perturbations on model predictions.

- Detailed evaluation and documentation of model performance metrics.

```
trainer.evaluate()
```

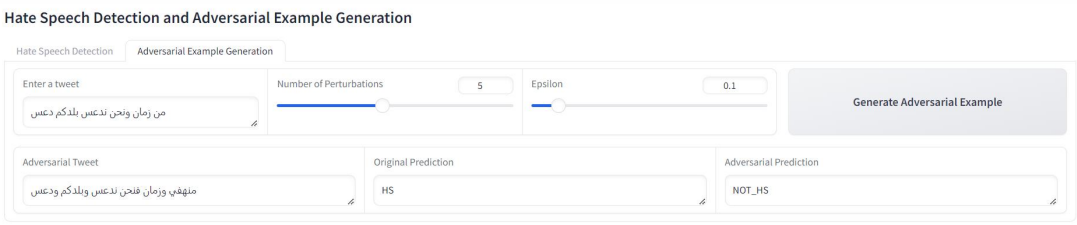
```
[80/80 00:06]
(1269, 2)
(12, 1269, 12, 64, 64)
1269
              precision    recall  f1-score   support

           0       0.95      0.99      0.97      1160
           1       0.75      0.43      0.55       109

    accuracy                           0.94      1269
   macro avg       0.85      0.71      0.76      1269
weighted avg       0.93      0.94      0.93      1269

[[1144   16]
 [  62   47]]
{'eval_loss': 0.49510034918785095,
 'eval_macro_f1': 0.7567722974699718,
 'eval_macro_precision': 0.8473110637289742,
 'eval_macro_recall': 0.7086997785510915,
 'eval_accuracy': 0.9385342789598109,
 'eval_runtime': 13.2633,
 'eval_samples_per_second': 95.678,
 'eval_steps_per_second': 6.032,
 'epoch': 3.0}
```

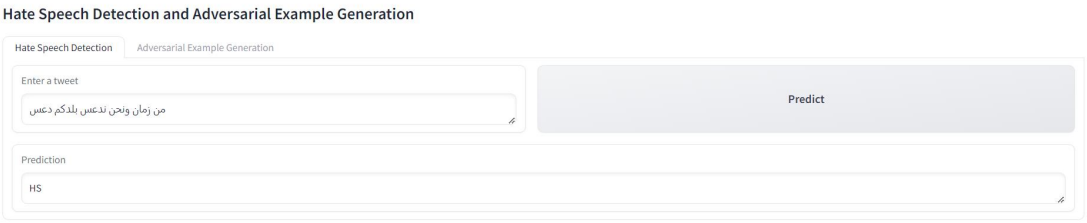
Robustness Analysis

- **Performance Against Attacks**: Show how the model performs when subjected to adversarial attacks, comparing results with and without adversarial defenses.
- **Error Analysis**: Provide an error analysis, highlighting common mistakes made by the model and any patterns observed in the misclassified examples.
- **Model Improvements**: Discuss the improvements observed after implementing defense strategies, focusing on the increase in robustness and overall model reliability.

GUI (web application):

The result of the adverserial attack:





Key Findings:

- The BERT-based Qarib model demonstrates superior performance compared to traditional methods.
- Preliminary results highlight the importance of cultural context in detecting offensive content.
- Initial perturbation analysis indicates areas for improvement in model robustness.

Challenges and Solutions:

- **Challenges**: Identify the key challenges faced during the project, such as handling the complexities of Arabic text, the computational cost of adversarial training, or difficulties in balancing model accuracy with robustness.

Solutions : Explain the solutions you implemented to overcome these challenges, highlighting any innovative approaches or adjustments to existing methods.

- **Challenge:** Handling the diverse linguistic landscape of the Arabic language.
    - **Solution:** Focus on emojis and extensive preprocessing to normalize and clean text data.
- **Challenge:** Ensuring model robustness against adversarial attacks.
    - **Solution:** Implementing a comprehensive perturbation mechanism and planning for adversarial training.

Future Work and Improvements:

- **Model Enhancement:**Experiment with more advanced models, such as transformers or ensemble methods, to improve classification accuracy and robustness.

● **Further Robustness Testing:** Continue evaluating the model against adversarial examples to strengthen its defenses.
● **System Optimization:** Work on optimizing the processing pipeline to reduce latency and improve performance in real-time applications.
● **Enhanced Features:** Explore additional features for the Serbot, such as real-time monitoring and response adjustments based on detected content.

Gantt chart for this project involves outlining the tasks and their durations:

Task 1:

Data Preparation, preprocessing and data Augmentation

Task 2:

Model Setup and Loading: BERT-based Qarib Model and fastText Model and their Evaluation and performance (with fine tunning)

Task 3:

Adversarial Attack Method with testing

Task 4:

Evaluation and Analysis: Evaluate the model's robustness and document findings.

Task 5:

Project Enhancment

Task 6:

Project Web application with gradio GUI

Task 7:

Final Report showing all the imporovment and Evaluation

Gant Chart: