



Prepared by group 7

THE EXCUSINATOR

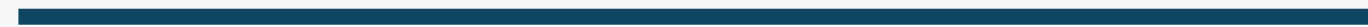
JUNE 12, 2025



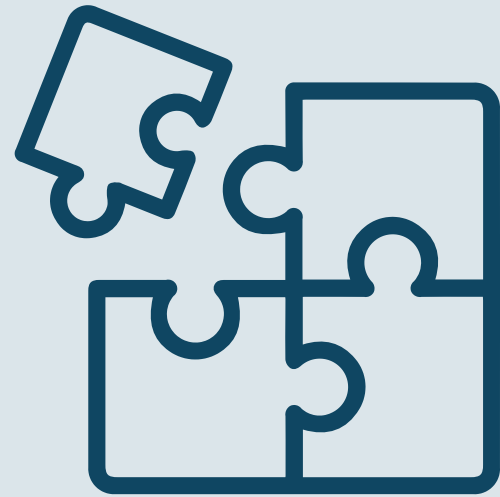
Problem statement



Build a model to classify student homework excuses into three different categories: plausible, funny and fake using logistic regression



Project Phases



Collecting Excuses

- Collecting real excuses from students using google forms
- If there are insufficient responses, we will use online resources.



Sorting The Database

- Labelling the excuses into categories
- Compiling all the excuses into a single database



Implementing The Classifier

- Code the logistic classifier and implement it

Technical specifications

Custom dataset

- Around 200 excuses, 65 excuses in each category
- Each row contains an excuse and a label
- Obtained from our survey and found online

Libraries

- Scikit-learn - modeling, preprocessing, evaluation
- pandas - data loading and manipulation
- joblib - saving/loading model weights
- matplotlib & seaborn - visualization

My cat ate my homework	plausible
My hamster chewed the USB cable	plausible
I was tired and fell asleep	plausible
I had a migraine and couldnt study	plausible
I was in a car accident	plausible
I accidentally submitted a draft instead	plausible
I was procrastinating until last minute and didnt have time	plausible
I was getting married	plausible
I thought the due date was tomorrow	plausible
I didnt have engery to do it	plausible
I was on family vacation	plausible
My phone died mid-submission and the deadline had passed	plausible
My cat walked across my keyboard and deleted everything	plausible
My internet connection was unstable causing the upload to n	plausible
I accidentally submitted my grocery list instead of the assignr	funny
I wrote it as a rap and now it wont stop playing in my head	funny
I added so many memes that Google Docs crashed	funny
I wrote a roast of the assignment instead of doing it	funny
I started writing and ended up oversharing my childhood trauma	funny
I was explaining it to my friend and convinced myself it didnt n	funny
I finished it but couldnt find the submit button on my fridge	funny
I wrote it all in Pig Latin and now I cant translate it back	funny
I used too many emojis and lost track of the actual content	funny
I tried to cite my dog as a source and it got flagged as plagiaris	funny
I narrated it dramatically to my mirror and forgot to type it	funny

Model

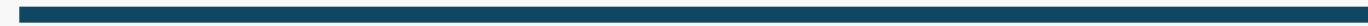
Logistic regression

- Classification algorithm used to predict whether an excuse is fake, funny, or plausible.

TF-IDF

- Convert text data (excuses) into numerical feature vectors.
- Assigns weights to words based on:
 - Term Frequency (TF): How often a word appears in a single excuse
 - Inverse Document Frequency (IDF): How unique a word is across the entire dataset
- Common words (the, and) get lower weights, while more distinctive words get higher weights

Interface



The Excusinator GUI provides a user-friendly interface for classifying homework excuses. The program uses Python built-in Interactive **Tkinter GUI** for the interface. Users can enter an excuse, view the predicted category with confidence scores, and confirm or correct the result.



Results

Input

The program requires an input and once the user clicks 'Predict', it will classify the input into a predicted category.



Excusinator

Enter Your Homework Excuse:

I lost my brother in an accident this week, so I could not work on my assignment.

Predict

Prediction: plausible

fake : 0.3044
funny : 0.2704
plausible : 0.4252

✓ Correct

✗ Incorrect

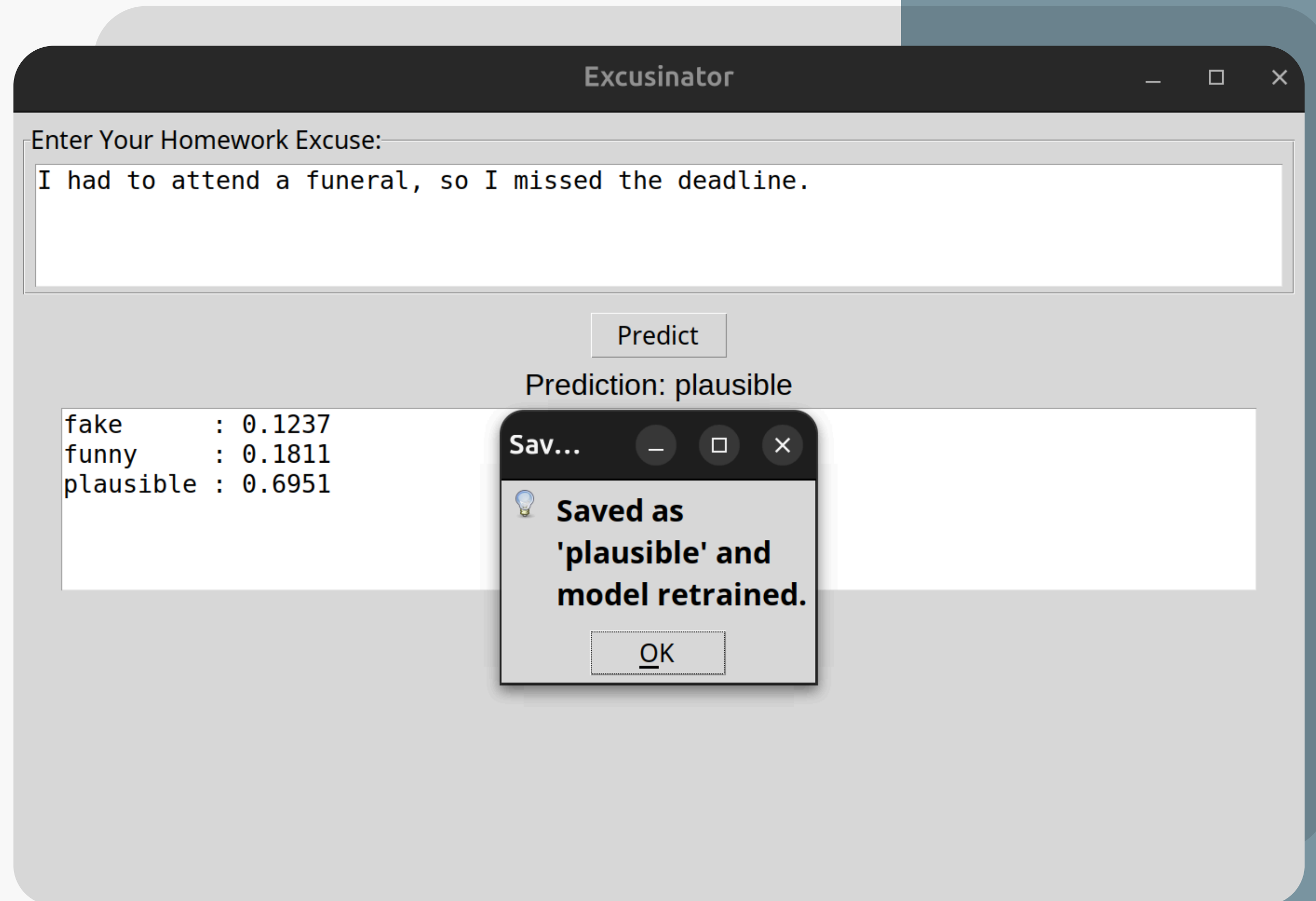
Results

Prediction Validation

After the initial prediction, program asks user if the prediction is correct.

Correct Prediction

In case the prediction is correct, the user can click on '✅ Correct'. That will save the input excuse and the predicted class to the database and retrain the model.



Results

Incorrect Prediction

In case the prediction is incorrect, the user can click on '✗ Incorrect'. That will show a dropdown list of classes that the user can choose as the correct class. Once chosen, the program save the input excuse and the chosen class to the database and retrain the model.



Excusinator

Enter Your Homework Excuse:

I had a bad feelin that submitting the homework would cause damage

Predict

Prediction: plausible

fake : 0.3298
funny : 0.2078
plausible : 0.4624

✓ Correct

✗ Incorrect

Select correct label

fake
funny
plausible

Evaluation

evaluator.py

A standalone script that evaluates the performance of the Excusinator. It loads the trained model (model_weights.pkl), TF-IDF vectorizer (vectorizer.pkl), and dataset (.csv file), then computes and displays:

Accuracy – overall correctness of predictions

Classification Report – precision, recall, and F1-score per excuse category

Confusion Matrix – visual heatmap showing how each excuse category is predicted

This script runs automatically and provides a clear summary of model performance without any user input.



```
joonayaka@uju:~/workspace/excusinator$ python evaluator.py
```

Classification Report:

	precision	recall	f1-score	support
fake	0.55	0.38	0.44	16
funny	0.60	0.60	0.60	15
plausible	0.44	0.64	0.52	11
accuracy			0.52	42
macro avg	0.53	0.54	0.52	42
weighted avg	0.54	0.52	0.52	42

Evaluation

```
joonayaka@uju:~/workspace/excusinator$ python evaluator.py
```

```
Classification Report:
```

	precision	recall	f1-score	support
fake	0.55	0.38	0.44	16
funny	0.60	0.60	0.60	15
plausible	0.44	0.64	0.52	11
accuracy			0.52	42
macro avg	0.53	0.54	0.52	42
weighted avg	0.54	0.52	0.52	42

Classification report (SciKit Learn)

- **Precision**

How many of the predictions were actually correct?

- **Recall**

How many of a certain label did the model correctly predict?

- **F1-score**

Mean of recall and precision

- **Support**

The number of instances for each label in the dataset

Evaluation

```
joonayaka@uju:~/workspace/excusinator$ python evaluator.py
```

```
Classification Report:
```

	precision	recall	f1-score	support
fake	0.55	0.38	0.44	16
funny	0.60	0.60	0.60	15
plausible	0.44	0.64	0.52	11
accuracy			0.52	42
macro avg	0.53	0.54	0.52	42
weighted avg	0.54	0.52	0.52	42

Classification report (SciKit Learn)

- **Accuracy:**

The percentage of total predictions that were correct

- **Macro avg:**

The average of precision, recall, and F1 across all classes, treating all classes equally, regardless of support.

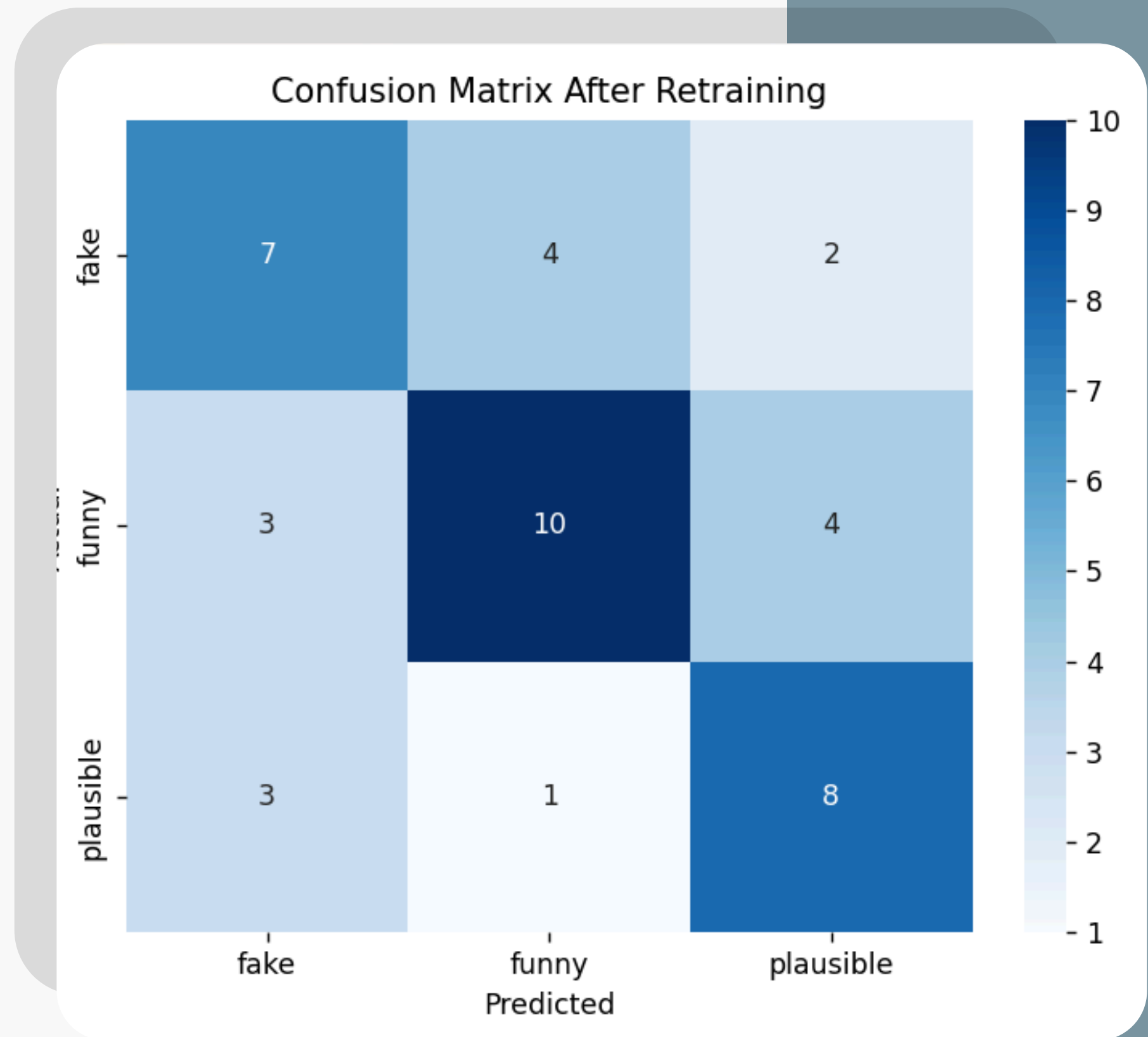
- **Weighted avg:**

The average of precision, recall, and F1 weighted by support

Evaluation

Confusion Matrix

Displays how many excuses of each category were correctly or incorrectly classified. Helps visually identify common misclassifications (e.g., confusing “funny” with “fake”) and diagnose weaknesses in the classifier.



Conclusion

The Excusinator is a robust and interactive machine learning application designed to classify student homework excuses into categories such as "funny," "plausible," "fake," and more.

- Using **TF-IDF vectorization** and logistic regression, it achieves reliable performance while remaining lightweight and explainable.
- The **intuitive GUI** allows users to test, confirm, or correct predictions in real time, with each interaction automatically updating the model through retraining.
- Evaluation metrics including **accuracy, precision, recall, F1-score**, and **confusion matrix** provide transparent insight into model performance.

Overall, the Excusinator demonstrates an effective, continuously learning classification system suitable for real-world educational or behavioral data applications.



THANK YOU!

