Fake News Classification

Jasleen Kaur (A20495939, jbhatia@hawk.iit.edu)

Pranjal Naik (A20489131, pnaik13@hawk.iit.edu)

Objective

In today's fast-paced digital world, the extensive spread of information has given rise to a significant challenge: the prevalence of fake news. This phenomenon not only undermines the credibility of media outlets but also has the potential to distort public opinion, and impact decision-making. It could also lead to the compromise of the integrity of democratic processes. The 'Fake News Classifier' project aims to tackle this critical issue and help create a more reliable information landscape for all.

Dataset

As per the dataset provided, we have 3 CSV files:

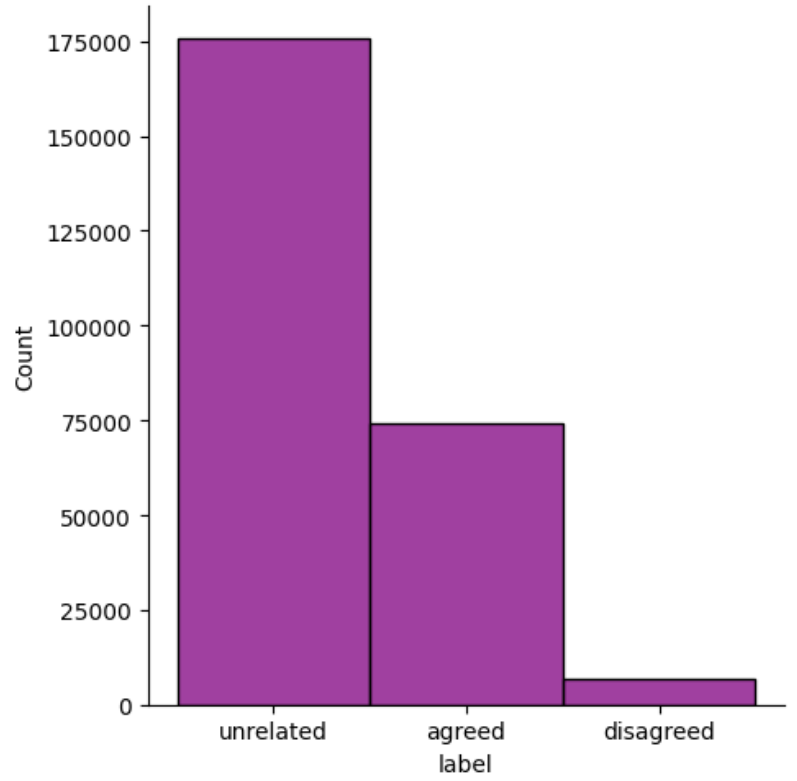
- **train.csv**: Training data,
- **test.csv**: Test data,
- **submission.csv**: Expected submission format

With columns as –

- **id**: the id of each news pair.
- **label**: indicates the relation between the news pair: agreed/disagreed/unrelated.

As we can see from the data distribution , the classes are quite uneven, making it exceedingly unlikely that any classification strategy will be effective or generalize effectively.

```
Records as per the label feature:
unrelated : 175598 (68.475)%
agreed : 74238 (28.949)%
disagreed : 6606 (2.576)%
```



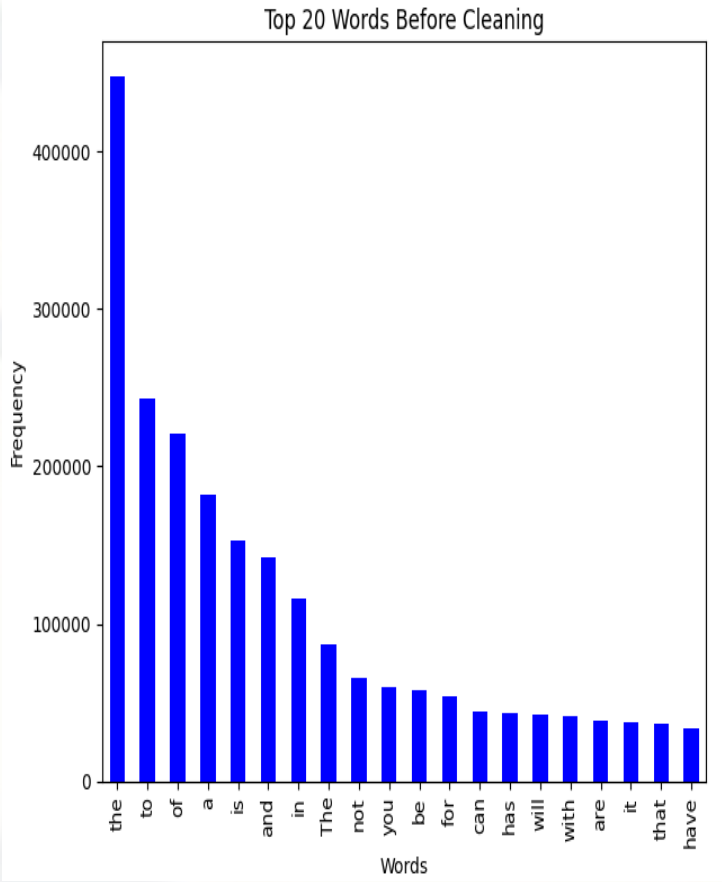
Data Preprocessing

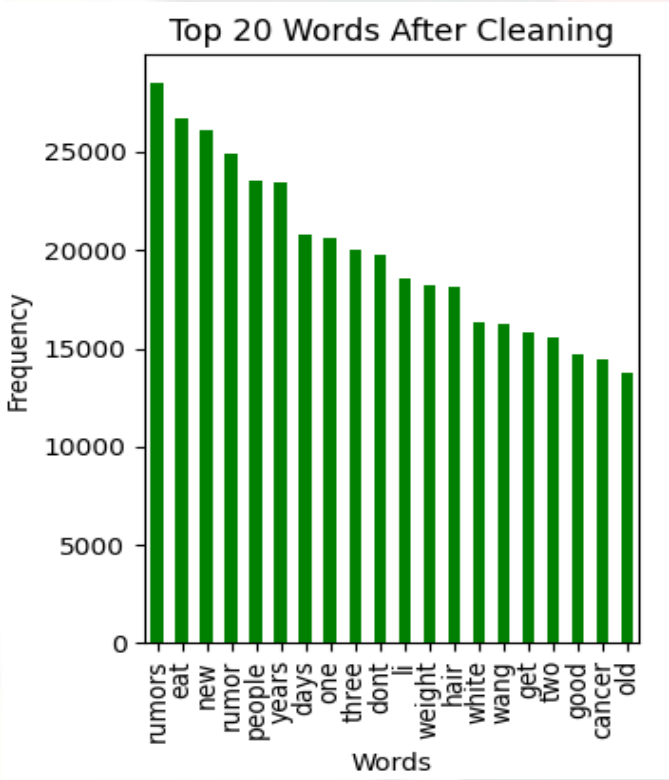
After we read the data from the files, we performed pre-processing on the data.

This includes –

- Removal of special characters and symbols
- Removal of stop words
- Normalization

The bar chart of the top 20 most frequent words in the text data before and after preprocessing are shown in the slide.



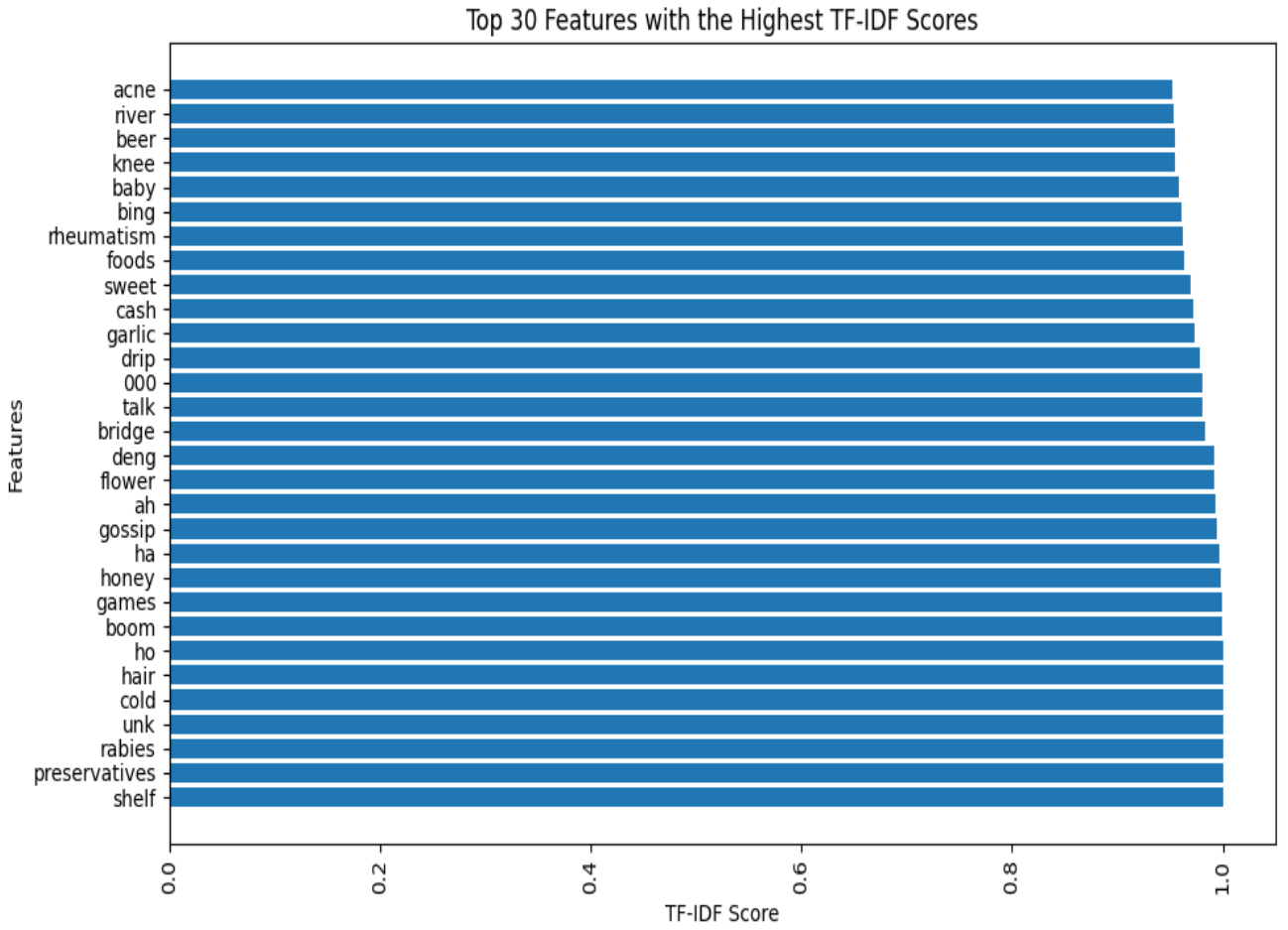


TF-IDF Vectorizer

We are using Term Frequency-Inverse Document Frequency for -

- Converting data into vectors
- To measure the importance of the term in a document relative to its frequency.
- To reduce the impact of common words.

The bar chart of the top 30 features with the highest TF-IDF scores are shown in the slide.



To Balance Dataset

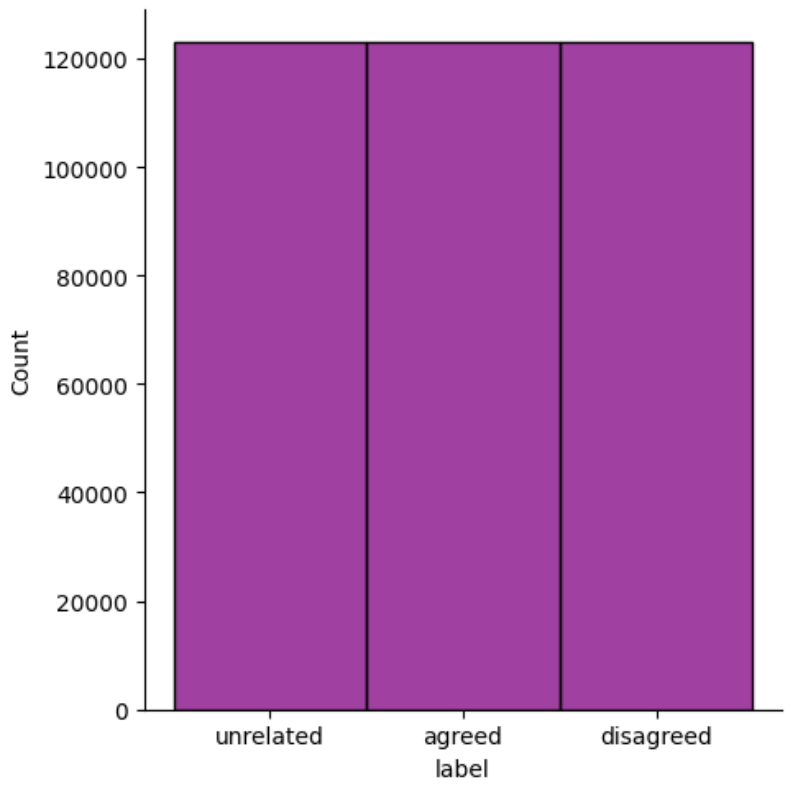
There are various methods to balance out the dataset. Most popular are -

- **Resampling the existing data**
- **Generate new text similar to the existing text.**

Here, We used SMOTE (Synthetic Minority Over-sampling Technique) for –

- Balancing imbalance datasets
- Improving the performance of the dataset

```
Records after balancing dataset as per label feature:
unrelated : 122918 (33.333)%
agreed : 122918 (33.333)%
disagreed : 122918 (33.333)%
```



To Evaluate Resampled Data

To determine whether oversampling the minority class using SMOTE has improved the model's performance on imbalanced datasets or not, we compared the performance of the model on both original and resampled datasets using cross-validation.

Here we used, 5-fold cross-validation (CV=5).

This helps to reduce the risk of overfitting and provides a more accurate estimate of the model's performance on unseen data.

```
Accuracy on original data:
0.795731681401021
Accuracy on resampled data:
0.8542144876771565
```

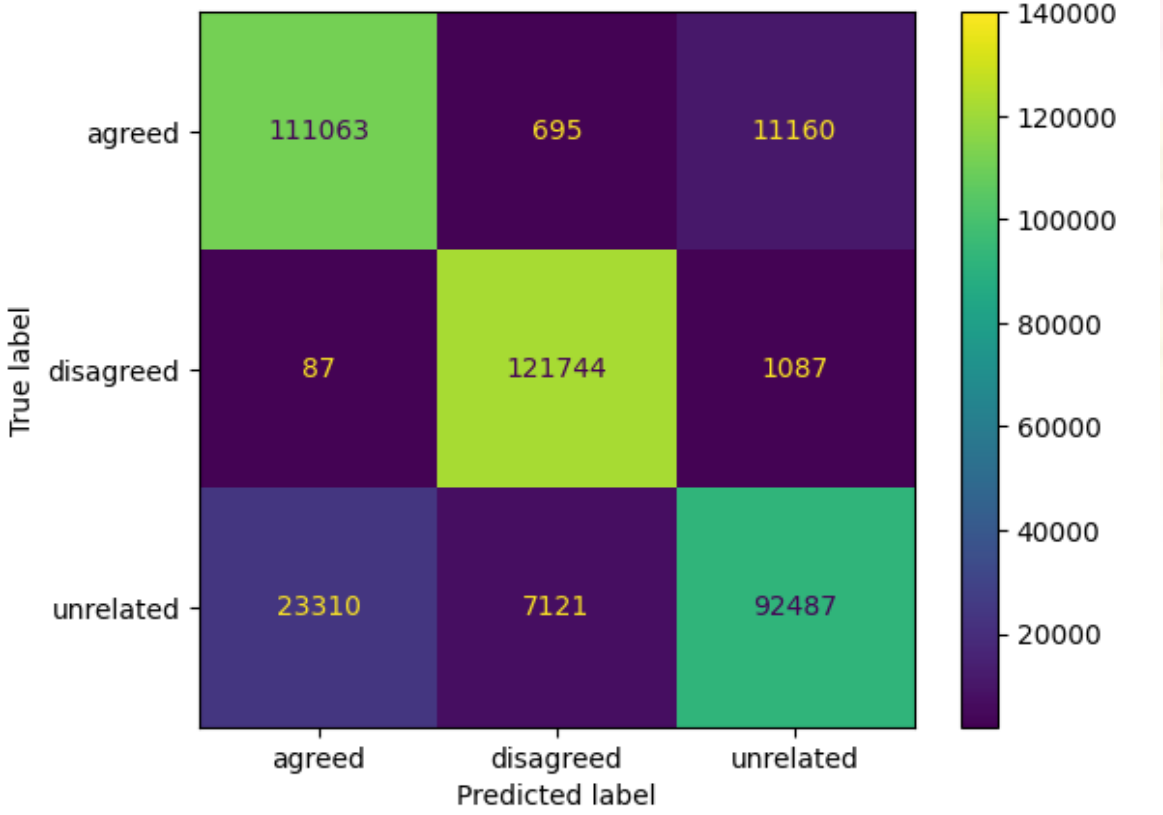
Logistic Regression

For Logistic Regression, We have used classifier as multinomial on both the titles.

The Training and Validation data is split into (70% and 30%)

The accuracy score and the confusion matrix of logistic regression is shown in the slide.

Logistic Regression Accuracy Score:
0.8821436513231042



Logistic Regression

- Classification report for validation data
- Advantages
  - Easy to implement
  - Less Training Time
  - Works well with small datasets
- Challenges
  - Unable to capture non-linear relationships
  - Limited interpretability
  - Not Generalized

```
Confustion Matrix for Validation Data:
[[13434    16  8821]
 [   52   370  1560]
 [ 4796    84 47800]]
Logistic Regression Accuracy Score for Validation Data:
0.8007487034172591
Classification Report for Validation Data:
              precision    recall  f1-score   support

      agreed       0.73      0.60      0.66     22271
   disagreed       0.79      0.19      0.30      1982
   unrelated       0.82      0.91      0.86     52680

    accuracy                           0.80     76933
   macro avg       0.78      0.57      0.61     76933
weighted avg       0.80      0.80      0.79     76933
```

```
The shape of training data is (256442, 6)
The shape of testing data is (64110, 5)
```

```
Training time for logistic regression: 88.5710 seconds
```

Results

We were successfully able to predict which news information from the dataset were fake or not.

These were labelled as “unrelated”, “agreed” ,and “disagreed”

Results are stored in .csv file using “.to_csv” function.

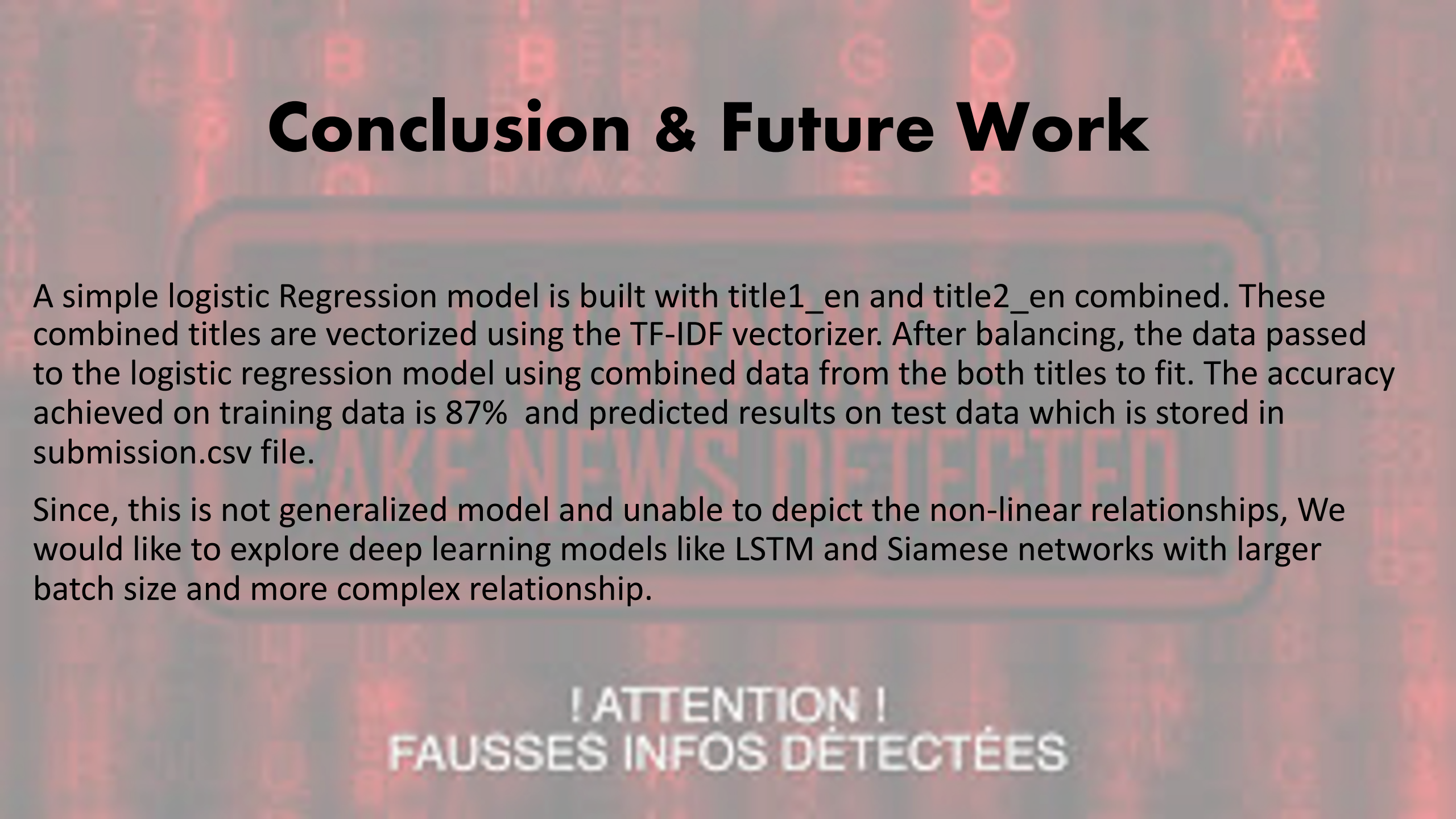
submission.csv

A	B	C
	id	label
0	256442	unrelated
1	256443	unrelated
2	256444	unrelated
3	256445	unrelated
4	256446	unrelated
5	256447	unrelated
6	256448	unrelated
7	256449	unrelated
8	256450	unrelated
9	256451	unrelated
10	256452	agreed
11	256453	agreed
12	256454	unrelated
13	256455	unrelated
14	256456	unrelated
15	256457	unrelated
16	256458	agreed

Conclusion & Future Work

A simple logistic Regression model is built with title1_en and title2_en combined. These combined titles are vectorized using the TF-IDF vectorizer. After balancing, the data passed to the logistic regression model using combined data from the both titles to fit. The accuracy achieved on training data is 87% and predicted results on test data which is stored in submission.csv file.

Since, this is not generalized model and unable to depict the non-linear relationships, We would like to explore deep learning models like LSTM and Siamese networks with larger batch size and more complex relationship.



Thank You


! WARNING !
FAKE NEWS DETECTED
! ATTENTION !
FAUSSES INFOS DÉTECTÉES