FINAL PROJECT – REPORT
SUMATH CHANDRA KILLE

PART1: EDA AND FEATURE ENGINEERING

A. Loaded the dataset with the name df

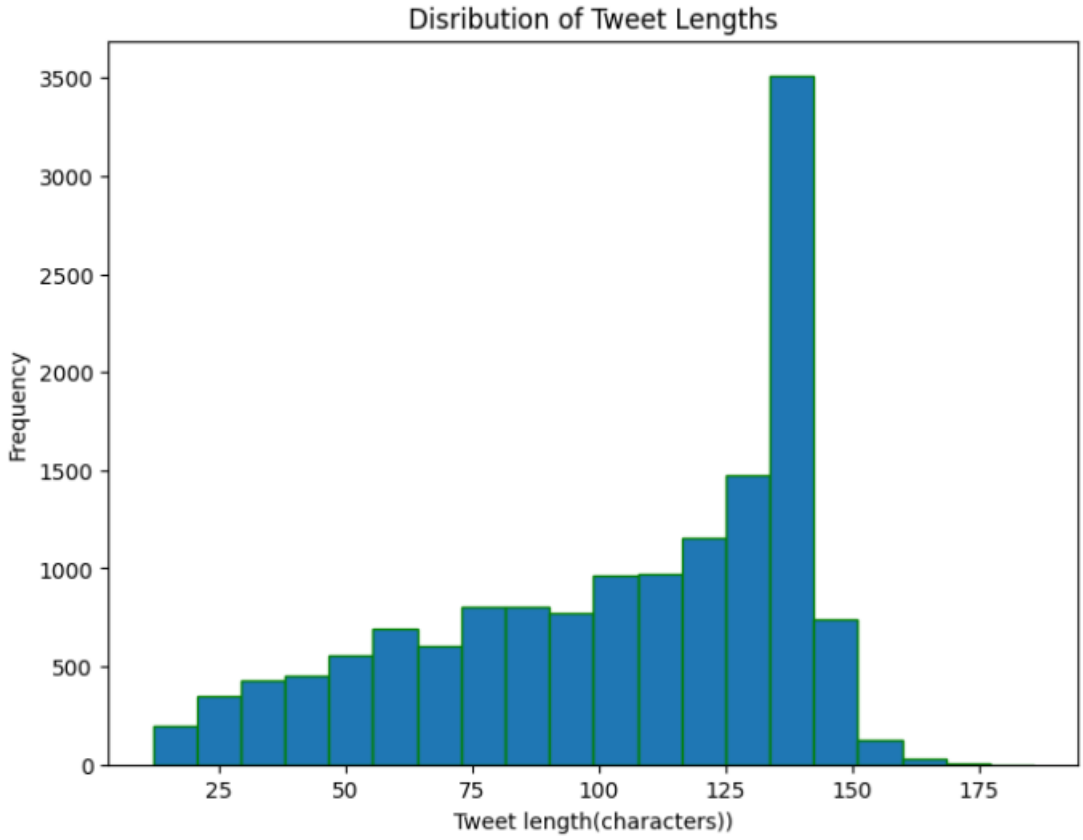
```
dataset shape: (14640, 15)
```

~Null values in the dataset:

```
tweet_id                      0
airline_sentiment             0
airline_sentiment_confidence  0
negativereason             5462
negativereason_confidence  4118
airline                       0
airline_sentiment_gold    14600
name                          0
negativereason_gold       14608
retweet_count                 0
text                          0
tweet_coord               13621
tweet_created                 0
tweet_location             4733
user_timezone              4820
dtype: int64
```

~As we are not using any other columns in model training except airline_sentiment and text which doesn't have any null values…I am not removing any null values from the other columns.

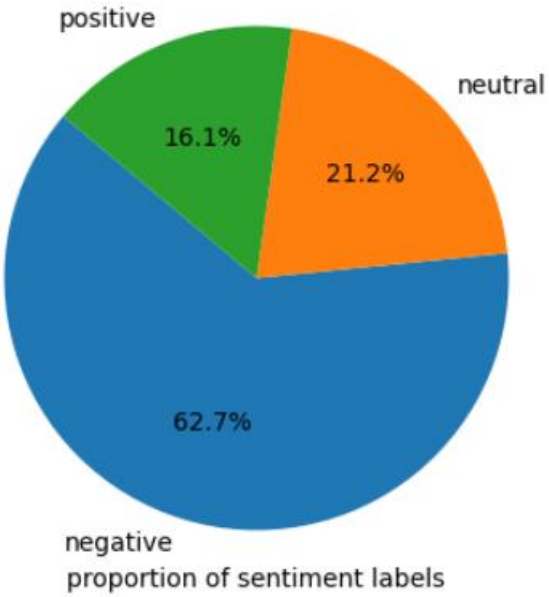
~Tweet length distributions graph:



~Observations: Most tweets fall in the 100-140 char range.

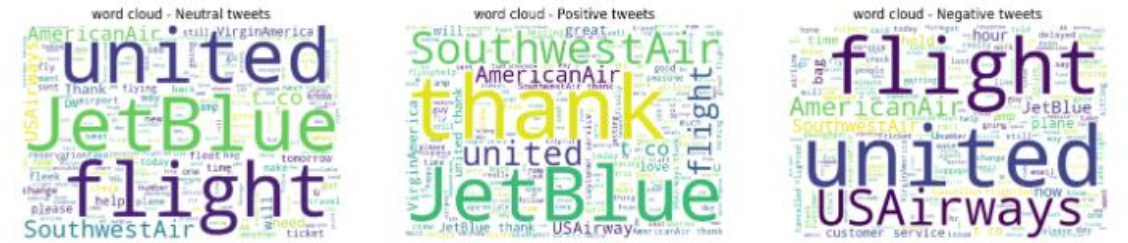
~Label proportions for the column airline sentiment:

Observation: Most of the tweets are negative, looks like slightly imbalanced, which may create impact in model training.

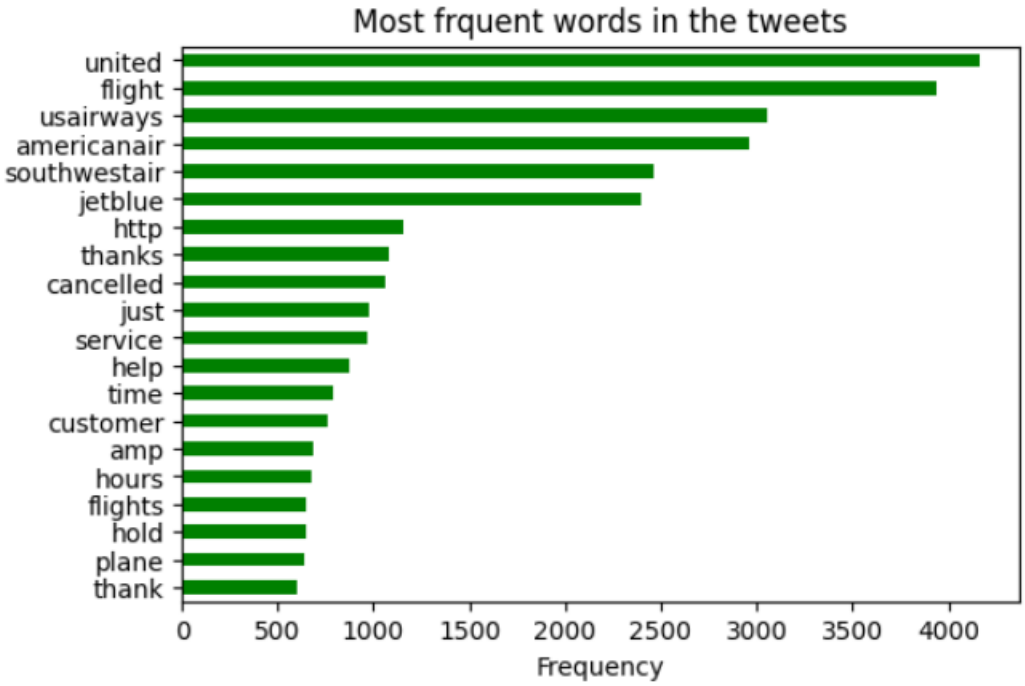


B. Text Visualization:

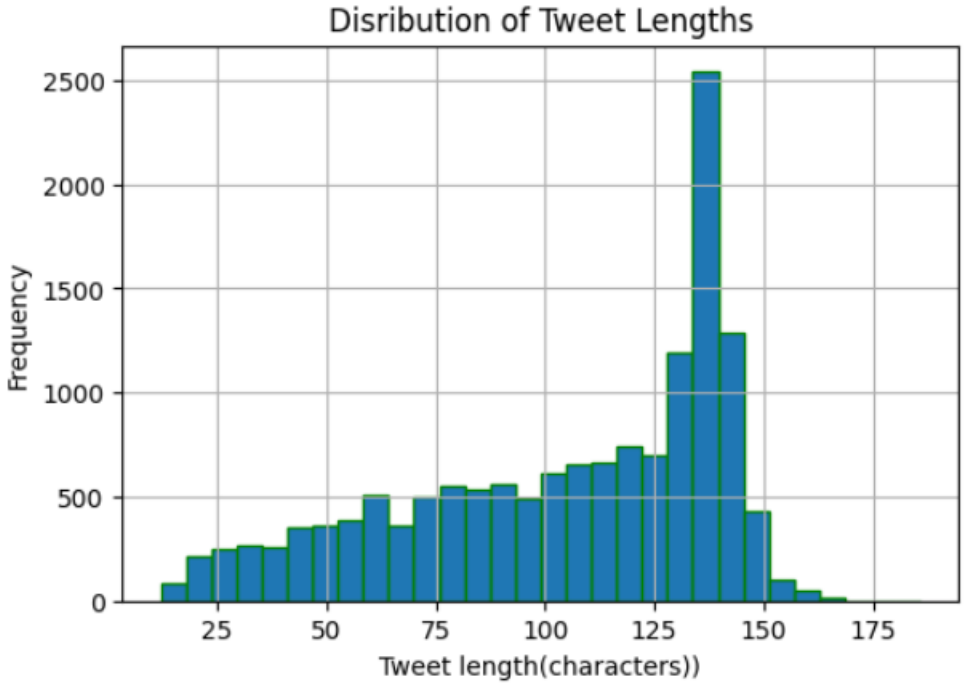
~Word count per class: plotted wordclouds for each class in the column airline_sentiment



~Top N frequent recurrence: using countvectorizer I plotted top 20 frequent words recurrence from tweets.



~Tweet/post length histograms: plotted histogram for tweet lengths using the text column.



C. Preprocessing and Feature Engineering:

~For text column I did lowercasing, punctuation removal, stop words, stemming and displayed top 5 results.

	text	cleaned_text
0	@VirginAmerica What @dhepburn said.	virginamerica dhepburn said
1	@VirginAmerica plus you've added commercials t...	virginamerica plu ad commerci experi tacki
2	@VirginAmerica I didn't today... Must mean I n...	virginamerica today must mean need take anoth ...
3	@VirginAmerica it's really aggressive to blast...	virginamerica realli aggress blast obnoxi ente...
4	@VirginAmerica and it's a really big bad thing...	virginamerica realli big bad thing

~Bag of words: where each row corresponds to tweet and each column to a unique word.
bow shape: (14640, 10805)
~TD-IDF: It assigns weights based on the word frequency and uniqueness.
tfidf shape: (14640, 10805)

PART 2 – MODEL BUILDING:

A. Model training:

Model 1 and Model 2 are the simple models Logistic Regression and Decision tree models were picked because they were easy to understand and provided reliable baselines against which the neural models could be compared.

Coming to Model 3 I initially tried a standard LSTM model but it failed to identify all three sentiment classes, mostly predicting the dominant class. Then for the improvement of the model I used Bidirectional LSTM which reads input from both directions and better captures context.

Finally Model 4 a Simple RNN model, despite its simplicity it performed well. Especially well on a shorter text due to lightweight structure and ability to detect patterns in shorter sequences.

Trained and evaluated four different models using the same feature set. All models used 80 and 20 train test split to maintain class distribution. Below are the results for all the four models.

```
Classification Report Logstic regression:
              precision    recall  f1-score   support

           0       0.82      0.93      0.87      1835
           1       0.65      0.52      0.58       620
           2       0.79      0.58      0.66       473

    accuracy                           0.79      2928
   macro avg       0.75      0.68      0.71      2928
weighted avg       0.78      0.79      0.78      2928


Confusion matrix logistic regression:
 [[1707   89   39]
 [ 260  325   35]
 [ 117   84  272]]

Classification Report D tree:
              precision    recall  f1-score   support

           0       0.72      0.92      0.81      1835
           1       0.47      0.20      0.28       620
           2       0.70      0.50      0.58       473

    accuracy                           0.70      2928
   macro avg       0.63      0.54      0.56      2928
weighted avg       0.67      0.70      0.66      2928


Confusion matrix Dtree:
 [[1682   82   71]
 [ 463  124   33]
 [ 179   57  237]]

Classification Report BIdirectional LSTM:
              precision    recall  f1-score   support

           0       0.84      0.87      0.85      1835
           1       0.57      0.51      0.54       620
           2       0.64      0.65      0.65       473

    accuracy                           0.76      2928
   macro avg       0.69      0.67      0.68      2928
weighted avg       0.75      0.76      0.75      2928


Confusion matrix BIdirectional LSTM:
 [[1594  152   89]
 [ 225  315   80]
 [  86   81  306]]
```

```
Classification Report RNN:
              precision    recall  f1-score   support

           0       0.82      0.87      0.84      1835
           1       0.54      0.48      0.51       620
           2       0.65      0.61      0.63       473

    accuracy                           0.74      2928
   macro avg       0.67      0.65      0.66      2928
weighted avg       0.73      0.74      0.74      2928


Confusion matrix RNN:
 [[1590  176   69]
 [ 236  296   88]
 [ 113   73  287]]
```

PART B: ARCHITECTURE AND IMPLEMENTATION DETAILS.

Pipeline overview:
~ For logistic regression and Decision tree, I used TF-IDF vectorization using TfidfVectorizer with max features 5000.
~For BiLSTM and RNN, I used keras tokenizer with num_words set to 200 and applied pad sequences to crate the fixed length in the input.
~All neural models used Embedding layers to convert tokens into dense vectors.

Train/Test Strategy:
~ All models used 80/20 train test split with stratification to maintain class balance.
~ Neural models used a 10% validation split from the training data to monitor performance during the training process.

Model Architecture:

Model1: Logistic Regression:
Vectorization: TF-IDF
Implementation: LogisticRegression() from sklearn
Regularisation: L2 regularization(default)
Hyperparameters: iterations count set to 1000
Evaluation metrics: Accuracy, Precision, Recall, F1 Score, Confusion matrix.

Model2: Decision Tree:
Vectorization: TF-IDF
Implementation: DecisionTreeClassifier() from sklearn
Hyperparameters: max depth set to 10
Evaluation metrics: Accuracy, Precision, Recall, F1 Score, Confusion matrix.

Model 3: Bidirectional LSTM:
Vectorization: Tokenizer + Padded sequences
Embedding layer:
~ Embedding(input_dim = 20000, output_dim = 128)
~Learns 128 dimensional dense vector representation of words.
Bidirectional LSTM layer:
~Bidirectional(LSTM(64)) – Reads the sequence in both forward and backward to capture full context.
Dropout Layer:

~Dropout(0.5) - 50% of neurons are randomly disabled during training to help avoid overfitting.
Output Layer:
~Dense(3, activation = 'softmax')
~ produces a probability distribution for each of the three sentiment categories: Positive, Neutral, and Negative.
Training configuration:
~ Loss function: Sparse_categorical_crossentropy which helps to multi class configuration.
~ Optimizer: Adam
~ Batch Size: 64
~ Epochs: 10
~ Validation Split: 10% of training set.

Model 4: Simple RNN Architecture and training details:
Note: Everything same as Bidirectional LSTM except the main layer.
Simple RNN layer:
SimpleRNN(64) – Process sequences in forward direction only, maintain the short-term context.

PART C – INITIAL OBSERVATIONS:
~ Which models performed better and are the performance differences significant?

Bidirectional LSTM (76% accuracy, F1-score 0.68), and Simple RNN (74% accuracy, F1-score 0.66) were the top-performing models. Predictions from these models were balanced for every class. On the other hand, Decision Tree scored the worst (69.7% accuracy, F1-score 0.56), particularly with minority classifications, and Logistic Regression had slightly higher accuracy (78.6%), but it was biased toward the negative class.

~ How did simple/explainable models perform in comparison with black-box models? Can you identify any area where simple models fail or performed poorly?

Although the decision tree and logistic regression models were simpler to learn, they had low F1-scores for positive and neutral sentiments and poor recall. Although having an overall F1-score of 0.71, logistic regression has trouble maintaining class balance. Many neutral/positive tweets were incorrectly identified by Decision Tree. Black-box models, such as RNN and BiLSTM, on the other hand, performed better consistently across all classes, which made them more suitable for this data.

3. INTERPRETATION, ERROR ANALYSIS AND TRADE-OFFS

A. ERROR ANALYSIS:

Below are some misclassified sentiment for the text/tweets.

	Text data	True sentment	Predicted sentiment	Is Misclassifie
0	united past	1	0	Yes
6	united this link in your tweet goes to someones internal email gt httptcozksx79itdn probably one of your 3rd party it contracts	1	0	Yes
9	southwestair when will the flight resume i dons see it in the open schedule	0	1	Yes
10	americanair can you check on he status of my exp membership card need the physical card to access some international lounges soon	1	0	Yes
16	americanair yes i will do just that as soon as i have a moment to gather my thoughts	1	0	Yes
18	usairways well i did miss it but gate agents had rebooked boarding pass waiting when i landed time for lunch amp a beverage easy cheesy	2	0	Yes
20	southwestair tried to rebook online but it says that i have to pay 200 for difference in price please help	0	1	Yes
27	americanair is there any way you could put me up in a hotel for the night if not im having to sleep at the airport again tonight	0	1	Yes
29	jetblue offering special fares to gopuregrenada discovergrenada httptcombcehcszx3 caribbejan islandexpert httptco2zm4jkalzl	1	0	Yes
30	southwestair any site gmail facebook etc	1	0	Yes
33	jetblue such a bummer but i understand its a business deal thanks for answering me much less sad now	1	0	Yes
37	americanair i can dm it to you if you follow me	1	0	Yes
39	southwestair julgood1 she was traveling with me the one that got miscommunicated with	0	1	Yes
46	southwestair can i get any kind of update on the delayed flight from boston to houston at 730 really need to be back home tonight	0	1	Yes
49	americanair a school trip of 38 including myself had to sleep over in the airport and are all on different standby flightsthis is not good	0	1	Yes

~ **Discuss ambiguities in the text and patterns in errors:**

~ Short tweets: Extremely brief text like “united past” lacks context, making sentiment hard to detect. it’s true sentiment is neutral but classified as negative.

~ Sarcasm or indirect tone: “jetblue such a bummer but i understand its a business deal thanks for answering me much less sad now” subtle language used in this kind of tweets which made model confused, and misclassified neutral message.

~ Class Confusion: “americanair can you check on he status of my exp membership card need the physical card to access some international lounges soon” This tweet is a polite request type of neutral sentiment, but the model interpreted as a complaint and made it a negative sentiment.

~politeness making negativity: “southwestair tried to rebook online but it says that i have to pay 200 for difference in price please help” Here customer is not satisfied with service and expresses negativity but the model misread as neutral due to their formal tone.

B.Case sensitivity and confusion matrix.

Analyse per-class precision, recall, and F1-score:

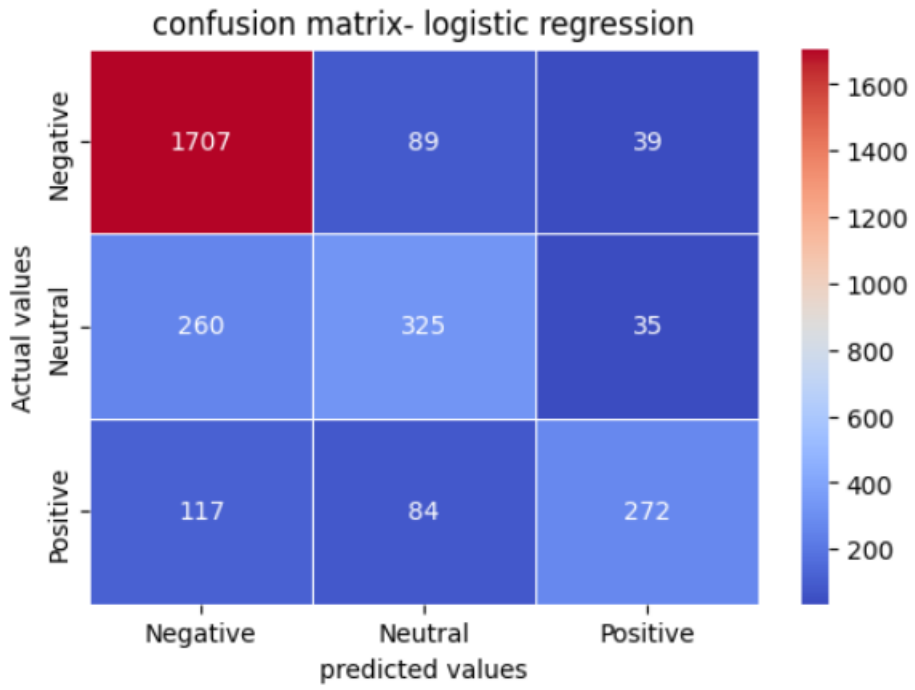
~Logistic Regression: Very strong performance on Negative, and much better balance for Neutral and positive than the Decision tree, but not more balanced than black box models.

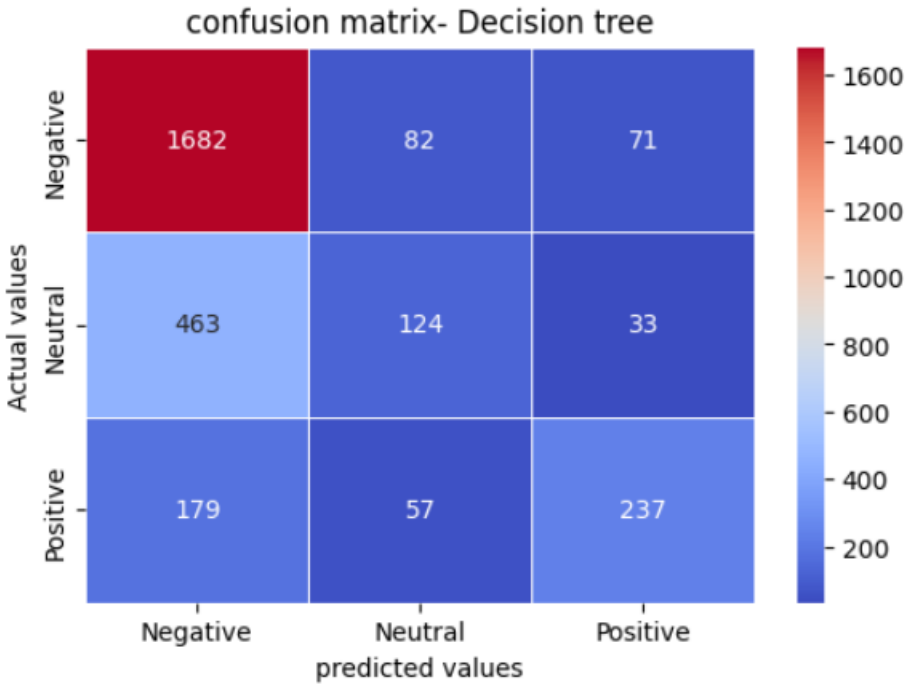
~Decision tree: Strong negative prediction, but very weak recall for Neutral, leading to poor class balance.

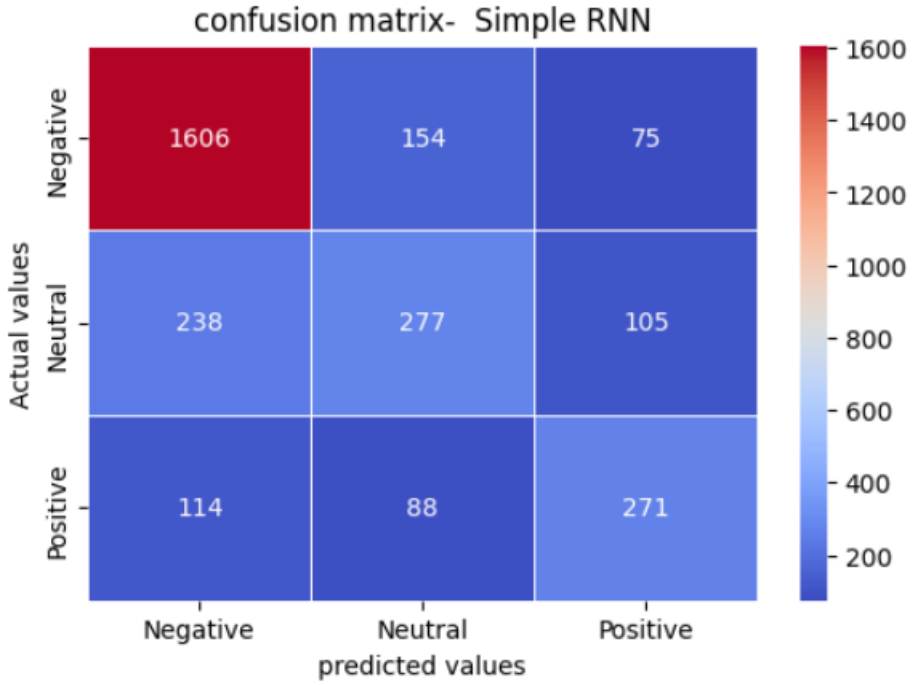
~ BiLSTM: Best balanced performance across all the classes, strong generalization and context understanding.

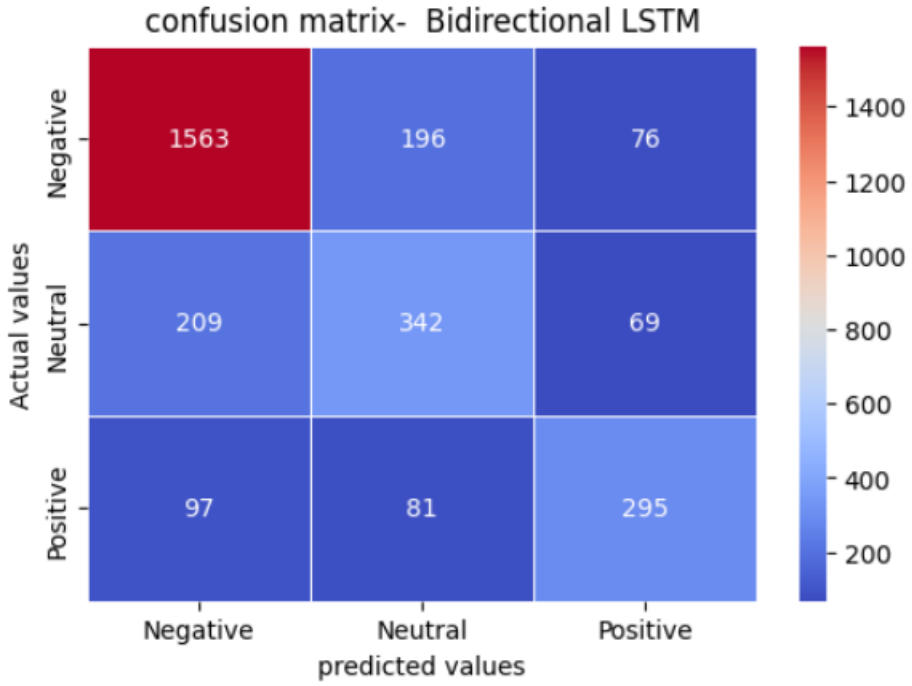
~ RNN: Simialar to BiLSTM, slightly behind in neutral class recall but still robust across all classes.

Visualize confusion matrices for all models:









~Discuss how class imbalance or text structure might impact model performance:

Impact of class imbalance: The dataset has high number of negative tweets, which causes models particularly the more basic ones like Decision Tree and Logistic Regression, are biased to predict the negative class. The confusion matrices make this clear, since positive and neutral tweets are frequently mistakenly categorized as negative.

Decision tree: For example, misclassified 463 neutral weights as negative.

Logistic regression: low recall on positive and Neutral labels.

Impact of tweet structure: The short, context limited, and informal tweets affects all the models, especially models like decision tree that rely heavily on the token presence, mixed-tone tweets will affect RNN-based models.

Ex: “please help me with fee” is one part of the tweet it is polite complaint, misread as neutral.

. **Explainability vs Performance: Comparison:**

- Compare explainable vs black-box models, is the performance gain from black-box models worth the loss in interpretability?

~ comparison: Black box models both achieved higher F1 scores and more balanced across all classes.

Explainable models (LR, DT) performed well on the negative class but struggled with Neutral and positive class.

Interpretability vs performance trade-off:

~Logistic regression is highly interpretable making it ideal for scenarios where explainability is critical like healthcare related domains. Decision tree provides rule based logic but it is overfitting on the text data.

~Both Black box models captured sentiment better especially in short, sarcastic tweets but they lack in explainability.

Conclusion:

Yes, the performance gain from black-box models worth the loss in interpretability in tasks like tweet sentiment classification, where handling language variation and class balance is more important than the transparency.

However, in high regulated environments, explainable models will be preferred.