DUPLICATE QUESTION IDENTIFICATION

STEPS:

1. We have around 4 lakh data points, therefore we have applied the normal algorithm for a random sample of 3000 data points.
2. To avoid imbalance we have taken the equal no of duplicate and not duplicate rows for the sample.
3. We have first tried to fit the data on 4 algorithms, the raw data without extra features.
4. Further to improve the accuracy we have tried to add on more features into the dataset.
5. We have tried with 8 features first and then added more 15 features too
6. Finally we have even tried various methods to create the matrix, methods like Bag of words(BOW), TD-IDF and Word2vec are used here.

The 4 classification algorithms used are:

1. Decision tree
2. Random forest classifier
3. XG boost
4. Logistic reg

Algorithms applied different forms, on dataset

1. Applied directly on the data
2. Added around 8 new basic features and checked different algorithms
3. Added around more 15 features : 8 + 15 features = 23 features added

Step 1:
Model directly on data to check accuracy

Models	Logistic Regression	Decision Tree	Random Forest Classifier	XG boost classifier
Accuracy	0.6722	0.6577	0.7375	0.7137

Step 2:
Improve accuracy by adding external features (around 8 new features)

Models	Logistic Regression	Decision Tree	Random Forest Classifier	XG boost classifier
Accuracy	0.757	0.7125	0.7763	0.7755

Compared to file 1 accuracies the accuracy for file 2 hs increased by adding on more features

Step 3: Adv feature added + text preprocessing too
Steps include:

1. Data preprocessing
2. Advanced features added
3. Used Bag of words, tf_idf, and word2vec for text representation
4. Used around 5-6 algorithms

1. DATA PREPROCESSING
    A) Lower case the texts
    B) Remove special characters, recurring words and pattern
    C) Edit contracting words
    D) Remove HTML tags
    E) Remove Punctuations

2. ADVANCED FEATURES
    A) TOKEN FEATURES eg: sachin is a great batsman ---> here wrds = 3, stop words = 2, token = 5
        1. cwc_min ----> no of common wrds/min(wrds(q1,q2))
        2. cwc_max -----> no of common wrds/max(wrds(q1,q2))
        3. ctc_min -----> no of common tokens/min(tokens(q1,q2))
        4. ctc_max -----> no of common tokens/max(tokens(q1,q2))
        5. csc_min -----> no of common stop words/min(stop words(q1,q2))
        6. csc_max -----> no of common stop words/max(stop words(q1,q2))
        7. last_word_equal ---> if both last wrd in both sent match then == 1, else 0
        8. first_word_equal--->if both first wrd in both sent match then == 1, else 0

    B) LENGTH BASED FEATURES
        1. mean_length ----->mean length of both the qs
        2. abs_length_diff--> find the difference in th both lengths and then take absolute of it
        3. longest_substr_ratio

    C) FUZZY FEATURES
        1. fuzz_ratio
        2. fuzz_partial_ratio
        3. token_sort_ratio
        4. token_set_ratio

3. TEXT REPRESENTATION - tried for various text representation methods
    A) Bag of words
    B) Tf_idf
    C) Word2Vec

With Bag of words

Models	Logistic Regression	Decision Tree	Random Forest Classifier	XG boost classifier
Accuracy	0.7376	0.713	0.7875	0.7866

With Td_idf

Models	Logistic Regression	Decision Tree	Random Forest Classifier	XG boost classifier
Accuracy	0.6995	0.7122	0.7907	0.7891

With Word2vec

Models	Logistic Regression	Decision Tree	Random Forest Classifier	XG boost classifier
Accuracy	0.7218	0.701	0.783	0.7812

Based on accuracy we can see XGBoost and random forest model works well compared to Decision Tree and Logistic regression.
We can use Random forest with TF-IDF text vectorization on this dataset, as the best model.
Further we can analyze the confusion matrix of the models with best accuracy.

Analyzing the confusion matrix:

In this dataset:
1 represents that sentence is duplicate
0 represents the sentences and not duplicate
Here the major problem will arise when we misclassify not_duplicate as duplicate
Meaning Actual = 0 and Predicted = 1
Hence we need to look for this error to be small in confusion matrix

Models	Random Forest Classifier	XGBoost Classifier
BOW	374	406
TF_IDF	340	402
Word2vec	479	529

As Random Forest classifier with TF_IDF representation of texts give a good accuracy and least False positive values, we select this as our best model