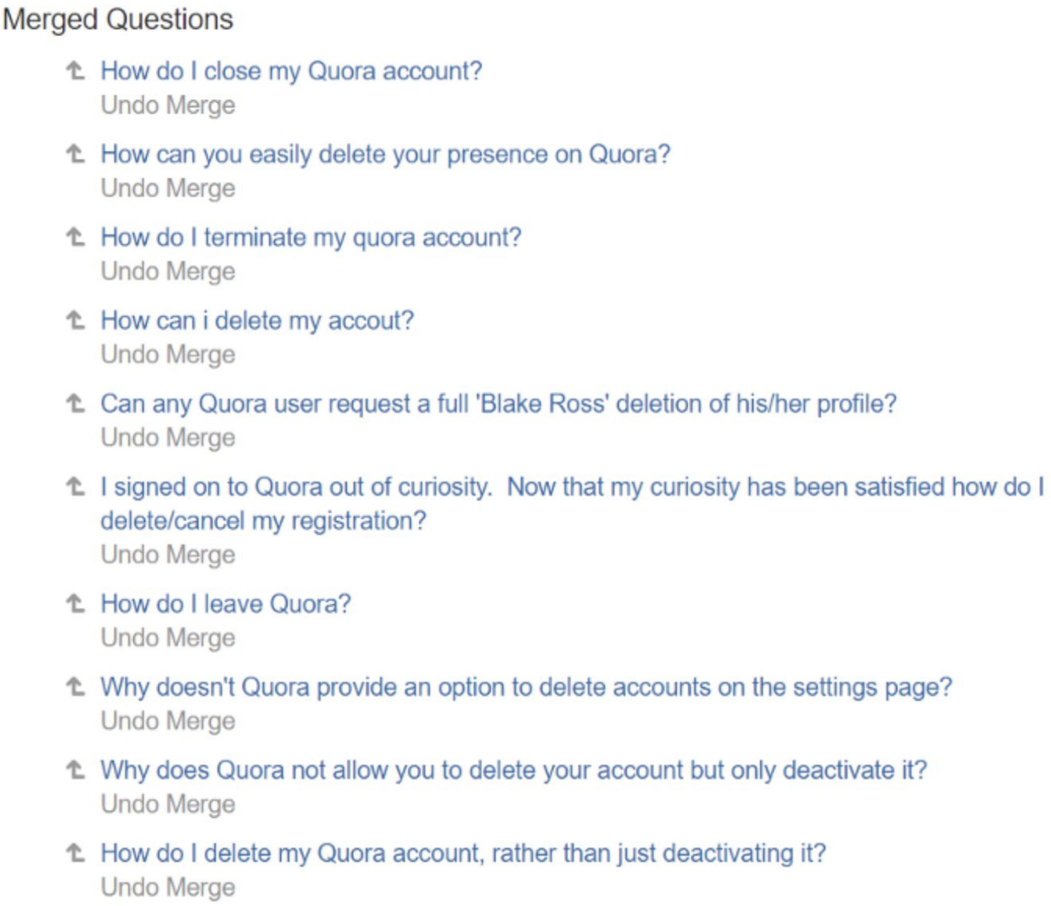
Content

- Problem Statement
- Dataset & Pre Processing
- Exploratory Data Analysis
- Feature Extraction
- Feature Analysis
- Data Rebalance
- Machine Learning Modeling
- Future Work

Problem Statement

- Identify which questions asked on Quora are duplicates of questions that have already been asked
- This could be useful to instantly provide answers to questions that have already been answered
- We are tasked with predicting whether a pair of questions are duplicates or not



Merged Questions

How do I close my Quora account?
Undo Merge

How can you easily delete your presence on Quora?
Undo Merge

How do I terminate my quora account?
Undo Merge

How can i delete my accout?
Undo Merge

Can any Quora user request a full 'Blake Ross' deletion of his/her profile?
Undo Merge

I signed on to Quora out of curiosity. Now that my curiosity has been satisfied how do I delete/cancel my registration?
Undo Merge

How do I leave Quora?
Undo Merge

Why doesn't Quora provide an option to delete accounts on the settings page?
Undo Merge

Why does Quora not allow you to delete your account but only deactivate it?
Undo Merge

How do I delete my Quora account, rather than just deactivating it?
Undo Merge

Dataset

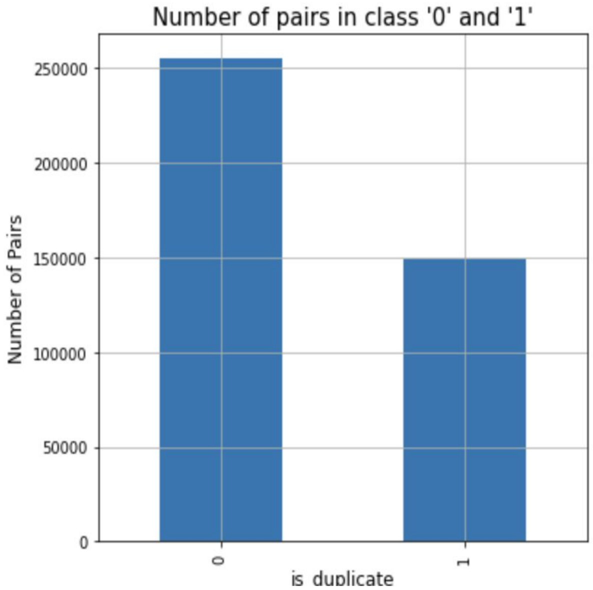
Variables

- Row Identifier
- Unique ID of each question pair
- Textual Contents
- Flag for ‘Is duplicate’ or not

Preprocessing

- Null Values Removal
- Convert text to lower case
- Removing HTML tags
- Removing punctuations
- Performing Stemming
- Removing Stopwords
- Expanding Contractions or decontract words
- Change abbreviations to its original terms
- Replace certain numerical values with strings (Eg: 1,000,000 with 1m)

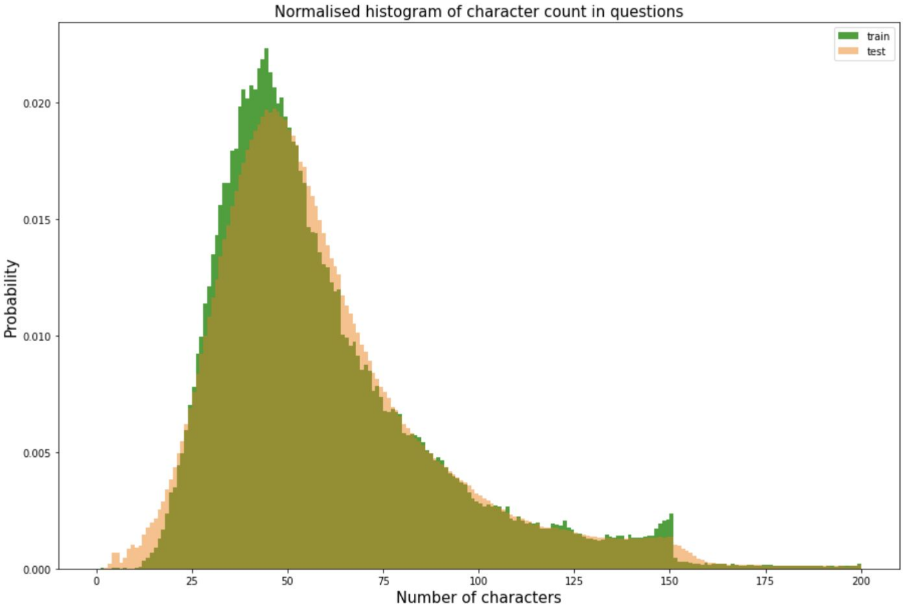
Exploratory Data Analysis



- Total number of question pairs for training: 404290
- Question pairs are not Similar (is_duplicate = 0): 63.08%
- Question pairs are Similar (is_duplicate = 1): 36.92%
- Total num of Unique Questions are: 537933
- Number of unique questions that appear more than one time:

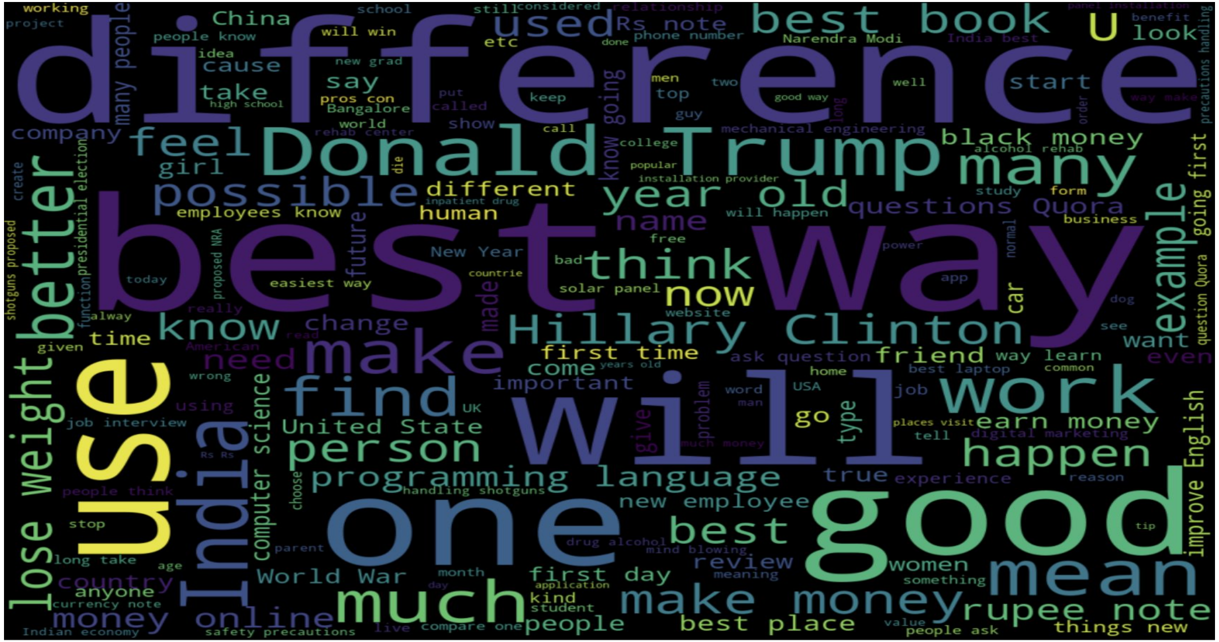
  111780 (20.77953945937505%)

Contd...

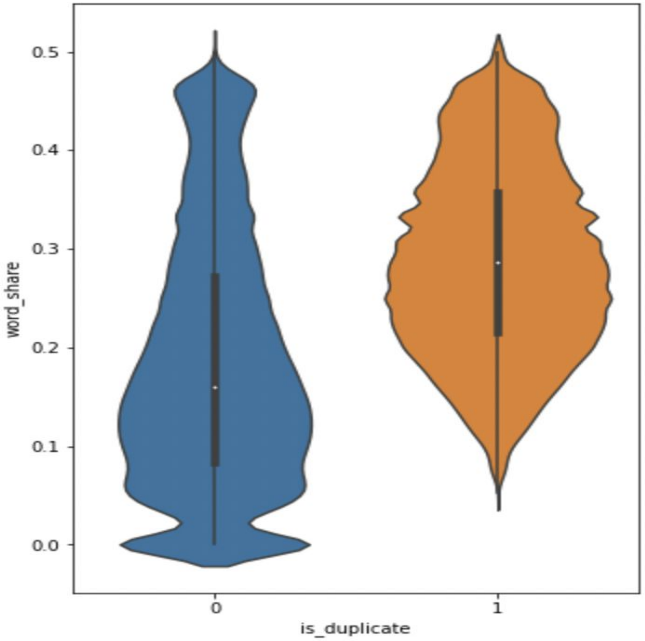


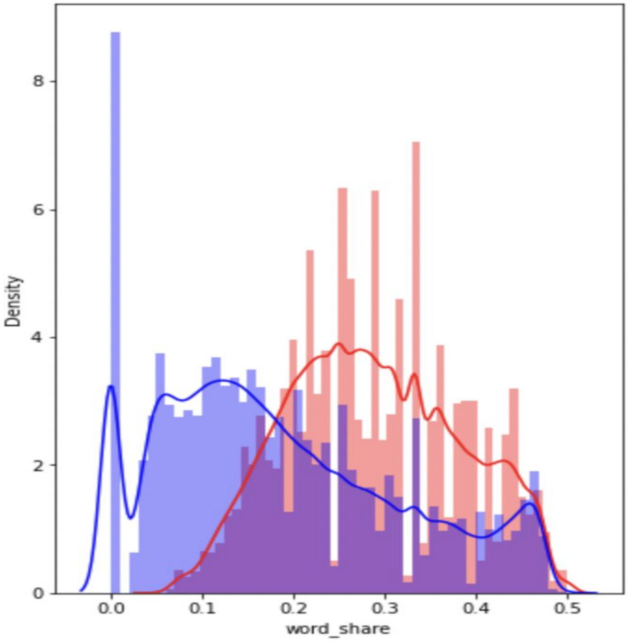
- Most questions have 15 to 150 characters in them
- test distribution is a little different from the train
- similar distribution for word count, with most questions being about 10 words long

Most common words



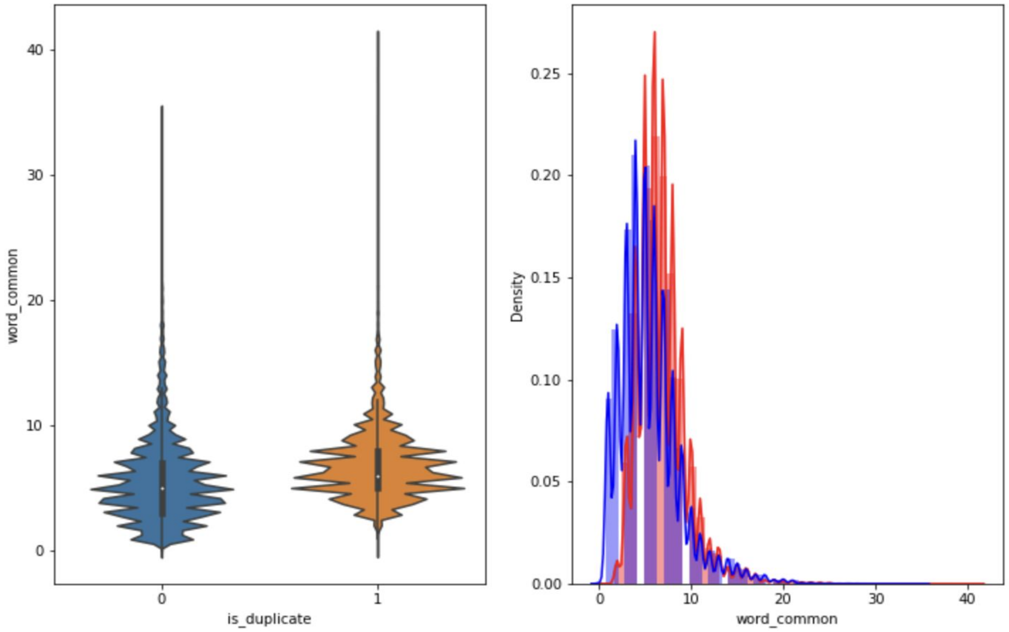
Feature Analysis





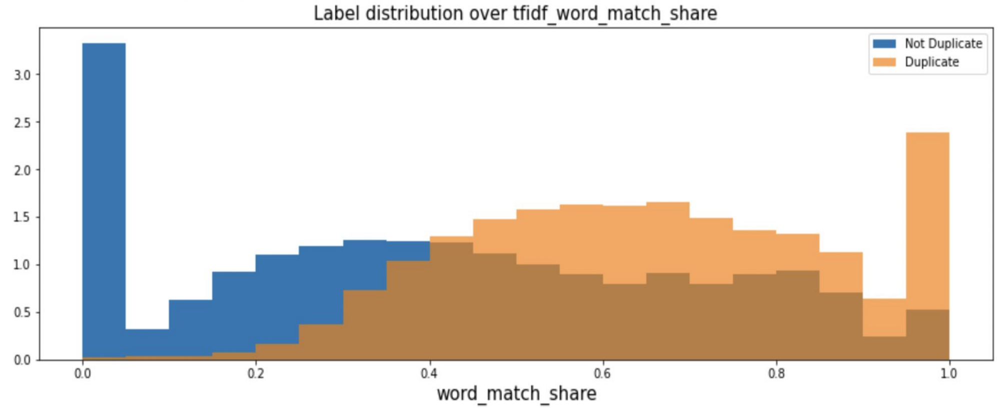
- The distributions for normalized word_share have some overlap on the far right-hand side, i.e., there are quite a lot of questions with high word similarity
- The average word share and Common no. of words of qid1 and qid2 is more when they are duplicate(Similar)

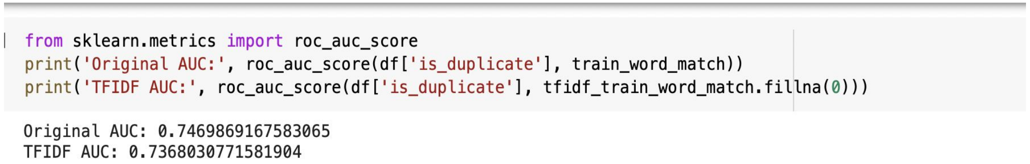
Contd...



The distributions of the word_Common feature in similar and non-similar questions are highly overlapping. Hence this feature cannot be used for classification.

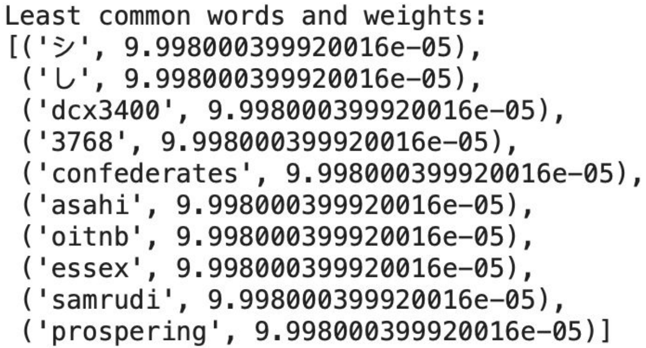
TF-IDF





```
from sklearn.metrics import roc_auc_score
print('Original AUC:', roc_auc_score(df['is_duplicate'], train_word_match))
print('TFIDF AUC:', roc_auc_score(df['is_duplicate'], tfidf_train_word_match.fillna(0)))
```

```
Original AUC: 0.7469869167583065
TFIDF AUC: 0.7368030771581904
```



```
Least common words and weights:
[('シ', 9.998000399920016e-05),
 ('し', 9.998000399920016e-05),
 ('dcx3400', 9.998000399920016e-05),
 ('3768', 9.998000399920016e-05),
 ('confederates', 9.998000399920016e-05),
 ('asahi', 9.998000399920016e-05),
 ('oitnb', 9.998000399920016e-05),
 ('essex', 9.998000399920016e-05),
 ('samrudi', 9.998000399920016e-05),
 ('prospering', 9.998000399920016e-05)]
```

Data Rebalance

Since we have 37% positive class in our training data, and only 17% in the test data. By re-balancing the data so our training set has 17% positives, we can ensure that XGBoost outputs probabilities that will better match the data

We have also oversampled the negative class to get better results.

Machine Learning Modelling: XGBoost

- A type of gradient boosting which gives weights to errors and provides faster results with better accuracy
- Test log loss: 0.39

Future Work

1. Advanced Feature Extraction using Fuzzy features and PoS tagging
2. t-SNE
3. Featurizing text data with weighted word vectors
4. Hyperparameter training
5. Neural Networks

Thank you!!

Questions?