DOCUMENTATION OF CODE

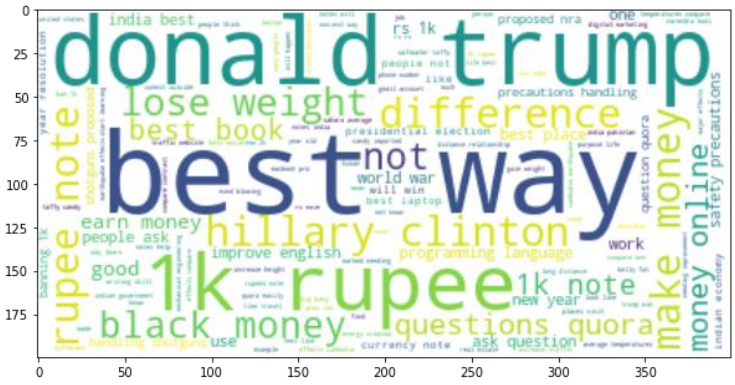
- Introduction:
  - We are working on the dataset of Quora question text similarity problem. We have used various methods for feature extraction, preprocessing of dataset, dimensionality reduction and evaluating on model.
  - We have implemented models like Logistic regression, Linear SVM and XgBoost.
- We have written code for evaluating the text similarity on the dataset obtained from the Quora repository.
- The dataset can be downloaded from the following link:
  - http://qim.fs.quoracdn.net/quora_duplicate_questions.tsv
- Read the dataset and analyze it.

```
#Dataset looks like
file.head()
```

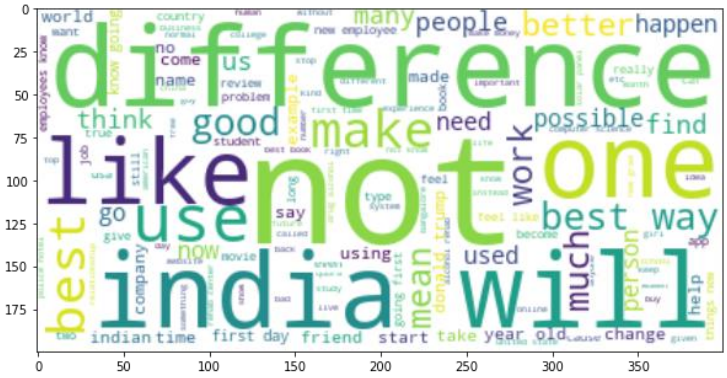
	id	qid1	qid2	question1	question2	is_duplicate
0	0	1	2	What is the step by step guide to invest in sh...	What is the step by step guide to invest in sh...	0
1	1	3	4	What is the story of Kohinoor (Koh-i-Noor) Dia...	What would happen if the Indian government sto...	0
2	2	5	6	How can I increase the speed of my internet co...	How can Internet speed be increased by hacking...	0
3	3	7	8	Why am I mentally very lonely? How can I solve...	Find the remainder when [math]23^{24}[/math] i...	0
4	4	9	10	Which one dissolve in water quikly sugar, salt...	Which fish would survive in salt water?	0

- Filled the null value with '1' in dataset.
- To make the model robust and very effective, calculated features before and after the preprocessing of dataset.
- Analyzed all the features.
- For extracting features, we need to split the sentence, for which we have used token.
- In preprocessing, stopwords are also removed using nltk.
- Plot the word cloud graph to understand our dataset better.

For Duplicate pair of question:



For Non-Duplicate pair of question:



- Used TSNE for analyzing high dimensional data and converting them into lower dimension data (both 2D and 3D)
- Added all the basic and advance features in one file and remove all the null values from dataset.
- Now the dataset of just features looks like:

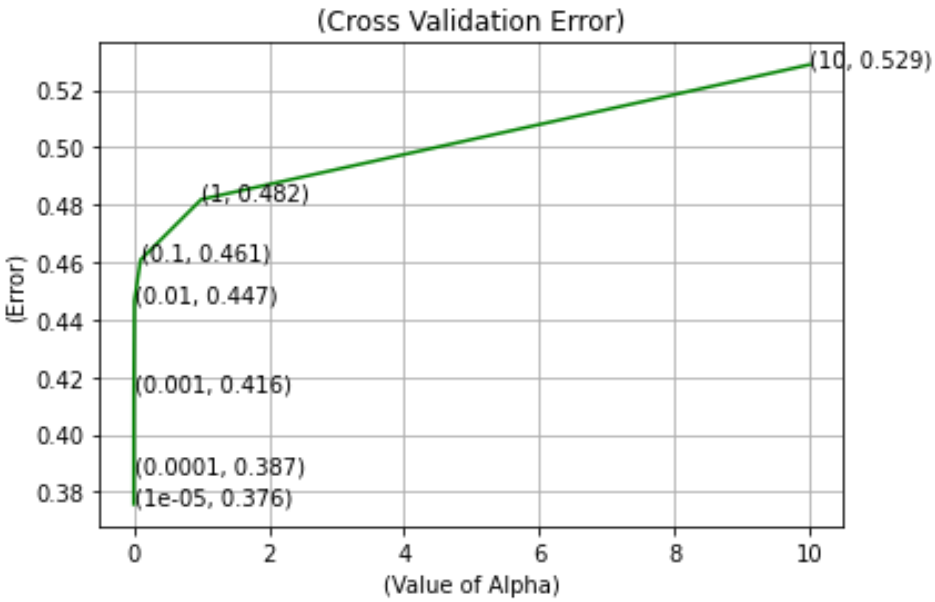
	id	freq_qid1	freq_qid2	q1len	q2len	q1_n_words	q2_n_words	word_Common	word_Total	word_share	freq_q1+q2	freq_q1-q2	cwc_min	cwc_max	csc_min	csc_max	ctc_min	ctc_max	last_wor
0	0	1	1	66	57	14	12	10.0	23.0	0.434783	2	0	0.999980	0.833319	0.999983	0.999983	0.916659	0.785709	
1	1	4	1	51	88	8	13	4.0	20.0	0.200000	5	3	0.799984	0.399996	0.749981	0.599988	0.699993	0.466664	
2	2	1	1	73	59	14	10	4.0	24.0	0.166667	2	0	0.399992	0.333328	0.399992	0.249997	0.399996	0.285712	
3	3	1	1	50	65	11	9	0.0	19.0	0.000000	2	0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
4	4	3	1	76	39	13	7	2.0	20.0	0.100000	4	2	0.399992	0.199998	0.999950	0.666644	0.571420	0.307690	

- Calculated TF-IDF vectorizer on both the columns ‘question1’ and ‘question2’ for the further training and testing purpose.
- Then split the dataset into train and test set in the ration of 70:30.
- Imported all the libraries required for the evaluation of final model.
- At first, we implemented logistic regression and calculated the log loss for different values of alpha as mentioned in the figure below.

Calculating the value of log loss for different value of alpha:

```
For alpha =  1e-05 The log loss is: 0.3757259506295624
For alpha =  0.0001 The log loss is: 0.3867683906140073
For alpha =  0.001 The log loss is: 0.4155543503526951
For alpha =  0.01 The log loss is: 0.4465394535292481
For alpha =  0.1 The log loss is: 0.46075642499166447
For alpha =  1 The log loss is: 0.48186916065055896
For alpha =  10 The log loss is: 0.5287824258455822
```

Plotting the graph “cross validation error”



The best value of log loss for train and test set for the best alpha value:

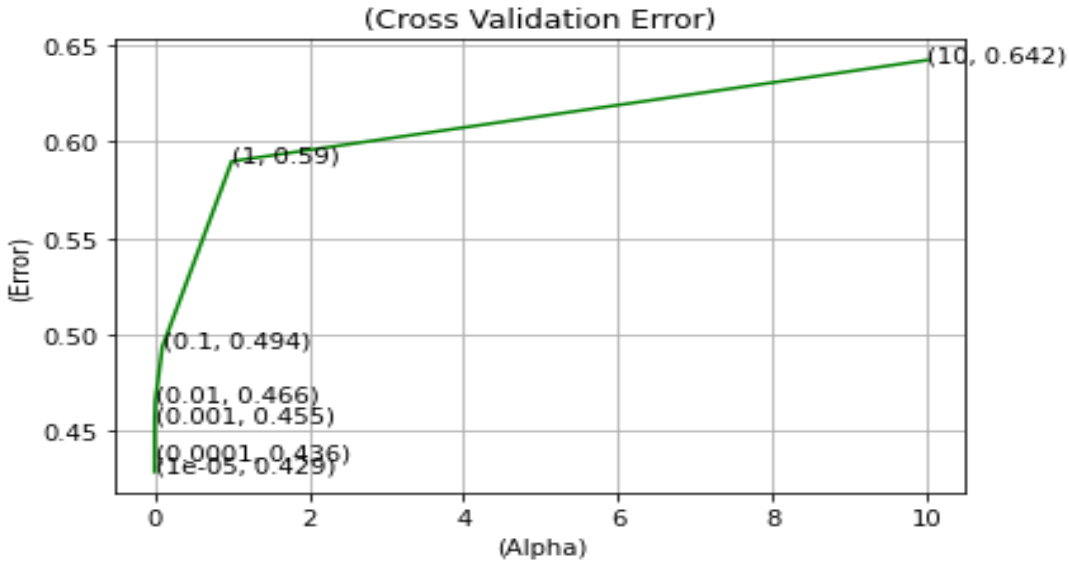
```
For the best alpha among all =  1e-05 The value of log loss for train is: 0.3733158407992717
For the best alpha among all =  1e-05 The value of log loss for test is: 0.3757259506295624
```

- Then we implemented linear SVM and calculated the log loss for different values of alpha as mentioned in the figure below:

Calculating the value of log loss for different value of alpha:

```
For alpha =  1e-05 The log loss is: 0.3757259506295624
For alpha =  0.0001 The log loss is: 0.3867683906140073
For alpha =  0.001 The log loss is: 0.4155543503526951
For alpha =  0.01 The log loss is: 0.4465394535292481
For alpha =  0.1 The log loss is: 0.46075642499166447
For alpha =  1 The log loss is: 0.48186916065055896
For alpha =  10 The log loss is: 0.5287824258455822
```

Plotting the graph “cross validation error”



The best value of log loss for train and test set for the best alpha value:

```
For the best alpha among all =  1e-05 The value of log loss for train is: 0.3733158407992717
For the best alpha among all =  1e-05 The value of log loss for test is: 0.3757259506295624
```

- The main difference between the logistic regression and linear SVM is that both uses different loss function. One uses ‘log’ and another uses ‘hinge’.
- The third and final model we implemented is XgBoost.
- As we have performed first linear SVM and logistic regression, hence we have called the file features file once again for XgBoost and performed necessary steps along with the train and test split steps again.
- The log loss calculated is as shown in the figure below:

```
The test log loss is: 0.3482895103792922
```

- Things worked and didn’t worked:
  - Using our best knowledge, we were able to implement everything with some decent challenges and some tough ones too.
  - One of the tough was to implement XgBoost, as we have to calculate vectors, which use trained GLOVE model (which is trained on Wikipedia). It caused the dimension problem in matrix, which was solved later by the guidance of professor and some online sources. But as it is too long to calculate, sometimes the session does timeout for long running.