Quora Question Pairs

1. Background and Information:

Quora is a place to gain and share knowledge—about anything. It's a platform to ask questions and connect with people who contribute unique insights and quality answers. This empowers people to learn from each other and to better understand the world.

Over 100 million people visit Quora every month, so it's no surprise that many people ask similarly worded questions. Multiple questions with the same intent can cause seekers to spend more time finding the best answer to their question, and make writers feel they need to answer multiple versions of the same question. Quora values canonical questions because they provide a better experience to active seekers and writers, and offer more value to both of these groups in the long term.

2. Data:

The dataset consists of Question IDs, Pairs of Questions and the Ground Truth (1 – questions are similar, 0 – questions are not similar). The ground truth labels have been supplied by human experts and are inherently subjective.

2.1. Features:

The data fields are described as follows:

- Id - the id of a training set question pair
- Qid1, Qid2 - unique ids of each question (only available in train.csv)
- Question1, Question2 - the full text of each question
- Is_duplicate - the target variable, set to 1 if question1 and question2 have essentially the same meaning, and 0 otherwise.

3. Target Business Question:

To mitigate the inefficiencies of having duplicate question pages at scale. Develop an automated way of detecting if pairs of question text actually correspond to semantically equivalent queries and also not to identify the syntactically same sentences as the duplicate ones.

4. Business Value:

An important product principle for Quora is that there should be a single question page for each logically distinct question. As a simple example, the queries "What is the most populous state in the USA?" and "Which state in the United States has the most people?" should not exist separately on Quora because

the intent behind both is identical. Having a canonical page for each logically distinct query makes knowledge-sharing more efficient.

This should not avoid posting a new question that has a minute difference from initial question syntactically but semantically varying a lot. For example, if there is a question like “What is a good method to invest?”. This question should not avoid the question like, “What is the good method to invest in USA?”. The distance between two sentences do not form a metric but the semantic sense between the sentences do.

5. Method:

5.1. Pre-processing:

The data provided was cleaned to remove the missing values and then preprocessed to remove insignificant data so as to make it ready for the analysis. The text is converted to lowercase, stopwords and punctuation were removed from the data as they form a part of the insignificant data that do not convey much about the meaning of the text. Examples of such words are by,or, the etc. The question mark at the end of each question was removed as it is making the last word as different one. For example, “invest? was not same as invest”.

5.2 Train and test data:

There were two sets of data given - train and test data. The train dataset given is split into training and validation data with 80-20 split ratio.

5.2. Proposed approach:

We have approached the problem in two ways. The first approach was to fit classification models with the similarity and distance measures between the question pairs as features. The second approach involves sentiment analysis using vectors generated from these question pairs. All error rates of all the models have been compared using a series of statistical tests. The proposed method covers the following topics:

- ➔ Sentiment Analysis
    - Doc2Vec
    - Word2Vec
    - Word2Vec + Tfidf
    - Bag-of-words + similarity measures
- ➔ Generalized Linear Model, the response variable is a binary data.
    - Logistic Regression
- ➔ Model intercomparison using data science experiments

6. Metrics:

6.1 Accuracy and mean of errors:

Accuracy is defined as the total number of correct predictions divided by the total number of predictions, where the predicted value is compared to the ground truth.

The mean of errors is considered to find the best model after a series of model comparison tests.

6.2 Logarithmic Loss:

Logarithmic Loss is a classification loss function that quantifies the accuracy of a classifier by penalizing false classifications. Minimizing the Log Loss is basically equivalent to maximizing the accuracy of the classifier.

$$l(y,p) = -y\log(p) - (1-y)\log(1-p)$$

where y is the ground truth and p is the predicted value.

This required a predicted probabilities of the model. SVM cannot give the probabilities, so we took accuracy as the major consideration for detecting the right model.

7. Prediction and results:

7.1. Prediction with similarity measures and vector distances:

7.1.1. Pre-Processing:

All mentioned pre-processing steps were performed. The question pairs were then converted to vectors using Doc2bow.

7.1.2. Features:

The features used for this method were the distance and similarity measures calculated using the above vectors. As, the questions are too small (Quora allows only 150 characters per question), taking the word similarity might work in favor of a few pairs. But taking a single similarity metric may not give a correct sense of sentence. So, the following metrics are considered.

7.1.2.1. Cosine similarity:

The cosine similarity between two vectors (or two documents on the Vector Space) is a measure that calculates the cosine of the angle between them. This metric is a measurement of orientation and not magnitude, it can be seen as a comparison between documents on a normalized space because we're not taking into the consideration only the magnitude of each word count of each document, but the angle between the documents.

7.1.2.2. Jaccard Coefficient:

The Jaccard coefficient measures similarity between finite sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets. It is computed as the number of shared terms over the number of all unique terms in both strings.

7.1.2.3. Euclidean Distance:

Euclidean distance of two documents X1 and X2 is defined as

$ED(X1, X2) = [(X1 - X2)(X1 - X2)\ T]^{1/2}$

The smaller the value of ED(X1, X2) is, the more similar the two documents are. We can see that this distance definition does not consider any patterns of term correlation that exist in the real-world data.

7.1.2.4. Manhattan Distance:

Manhattan distance of two documents X1 and X2 is defined as

$MD(X1, X2) = |(X1 - X2)|$

7.1.2.5. Minkowski Distance:

Manhattan distance of two documents X1 and X2 is defined as

$ED(X1, X2) = [[(X1 - X2)(X1 - X2)\ T]^{q}]^{1/2q}$

7.1.3. Prediction:

Using the above calculated metrics as features, we build three classification models to predict if the question pairs are similar. The classification techniques used were:

- Logistic Regression
- Decision Tree Classifier
- Support Vector Machine

7.1.4. Results:

Logloss and accuracy values of the classification models are shown below.

```
log loss values for logistic model 0.555795052746
log loss values for decision model 1.76042089047
```

```
Accuracy of logistic model 0.664
Accuracy of SVM model 0.673666666667
Accuracy of decision tree classifier model 0.691166666667
```

The Decision tree classifier performs better than that of the other two classifiers in terms of accuracy.

7.2. Logistic Regression with Word2Vec:

7.2.1. Pre-Processing:

All mentioned pre-processing steps were performed

7.2.2. Word2Vec:

Word2Vec is a method that captures the context of words, while at the same time reducing the size of the data. Word2Vec is a term used for similar algorithms that embed words into a vector space with 300

dimensions in general. These vectors capture semantics and even analogies between different words. The famous example is: 'King' – 'Man' + 'Woman' = 'Queen'. They can be used to compute semantic word similarity, classify documents, or input these vectors to Recurrent Neural Networks for more advance applications.

7.2.3. Prediction:

After the basic pre-processing steps, a Word2vec model was built to convert all the questions into vectors by taking the weighted mean of all word vectors present in each question. In line with the baseline method used, we find the cosine similarity between the word vectors of the question pairs. The similarity values and the ground truth for similarity are then fed into a regression model. This model helps us predict if the questions are similar based on the cosine distance measures.

7.2.4. Results:

Logloss and accuracy values of the logistic regression model with the confusion matrix is shown below.

```
accuracy:
0.635832199827
Confusion Matrix:
[[174397  29559]
 [ 88242  31282]]
Classification Report:
             precision    recall  f1-score   support

          0       0.66      0.86      0.75    203956
          1       0.51      0.26      0.35    119524

avg / total       0.61      0.64      0.60    323480

logloss:
0.614089675651
```

7.3. Logistic Regression with Word2Vec and TF-IDF:

7.3.1. Pre-Processing:

All mentioned pre-processing steps were performed

5.4.2 Term Frequency – Inverse Document Frequency:

Term frequency – inverse document frequency is a statistical measure used to evaluate how important a word is to a document in a corpus. The importance increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus.

5.4.3 Prediction:

Word2Vec is used to convert each question in the dataset into a semantic vector by taking the weighted mean of all word vectors of the words present in each question. The mean vector representation used in the above method is enhanced by using TF-IDF scores defined for each word. Weighing the word vectors using their TF-IDF scores helps in discriminating words and avoids useless, frequent words that are shared by many questions.

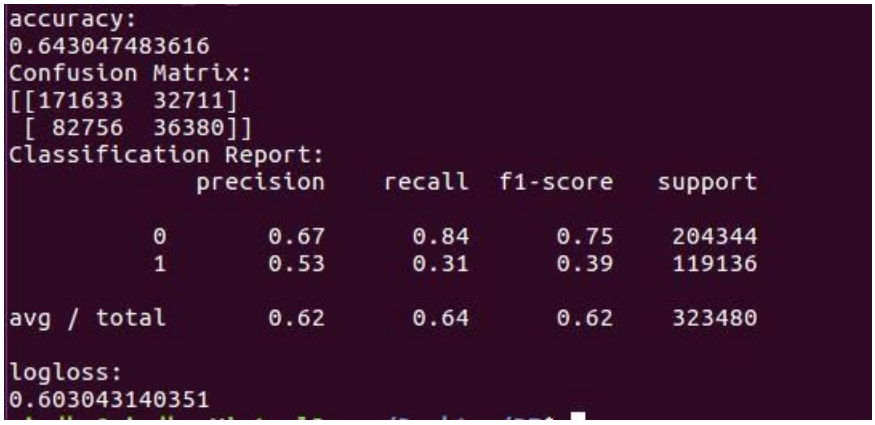
As the questions given in the data are limited which does not give sufficient information for the gensim Word2Vec model trained on this data to generate the word vectors, a pre-trained Word2Vec model

which comes free with Spacy is used. It is trained on Wikipedia and therefore, it is stronger in terms of word semantics. Similar to Gensim model, it also provides 300 dimensional embedding vectors.

Once the semantic vectors are obtained in the above mentioned manner for each question in the dataset, the similarity score is obtained by calculating the cosine similarity between the semantic vectors of two questions of the question pair. The similarity scores obtained indicate the extent to which the two questions are similar. The similarity scores of all the question pairs and their respective classification labels in the train data are used to train the model built using Logistic Regression. This trained model is then used to predict the label fopr each question pair in the test data given their similarity scores.

5.4.4 Results:

Logloss and accuracy values of the logistic regression model with the confusion matrix is shown below.



5.5 Logistic Regression with Doc2Vec:

5.5.1 Pre-Processing:

All mentioned pre-processing steps were performed.

5.5.2 Doc2Vec:

Doc2Vec is a variant of Word2Vec that is used when we deal with larger volumes of data. Similar to Word2Vec, there are two methods: Distributed Memory and Distributed Bag of Words. The former attempts to predict a word given its previous words and a paragraph vector. Even though the context window moves across the text, the paragraph vector does not and allows for some word-order to be captured. The later predicts a random group of words in a paragraph given only its paragraph vector. These paragraph vectors can be fed into a sentiment classifier without the need to aggregate words.

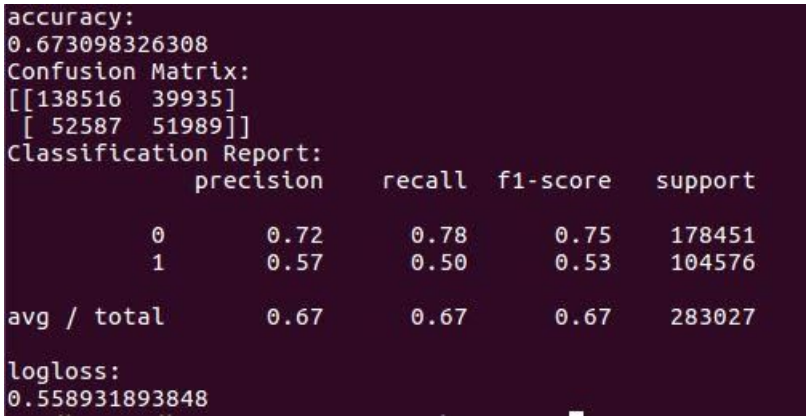
5.5.3 Prediction:

The Doc2Vec implementation available in gensim library is used to build a Doc2Vec model that is trained on the entire corpus. The question pairs in the entire dataset have been modified to generate labeled

sentences which have been used to train the Doc2Vec model. This doc2vec model is used to generate the feature vectors called paragraph vectors for each question in the dataset. The data is then split into training and test sets. For each question pair in the train data, the similarity score is obtained by calculating the cosine similarity between the paragraph vectors of two questions of the question pair. The similarity scores obtained indicate the extent to which the two questions are similar. The similarity scores of all the question pairs and their respective classification labels in the train data are used to train the model built using Logistic Regression. This trained model is then used to predict the label fopr each question pair in the test data given their similarity scores.

5.5.4 Results:

Logloss an accuracy values of the logistic regression model with the confusion matrix is shown below.



8. Model Comparison:

For each of the 3 models, that is, Doc2Vec, Word2Vec, Word2Vec+Tfidf, the error values have been generated by performing 10-fold cross-validation on the data. The following statistical tests are performed to compare the error values of different models on the dataset:

8.1 Normal Test:

scipy.stats.normaltest() is used to test whether the error values of each model follow a normal distribution

Null Hypothesis : The data follows a normal distribution.

Alternate Hypothesis : The data doesn't follow a normal distribution.

Result:

```
NormaltestResult(statistic=1.7925594301722056, pvalue=0.40808503177224031)
NormaltestResult(statistic=1.802613970022175, pvalue=0.40603862639126231)
NormaltestResult(statistic=3.235425615103571, pvalue=0.19835184951724982)
```

The respective p-values of three methods are, 0.408,0.406,0.198 respectively which are significant as they are greater than 0.05 which indicates that we fail to reject the null hypothesis for the three vectors. Hence, it is proved that all the three error vectors are normally distributed.

8.1 Barlett Test:

Bartlett's test is performed to check for equal variances among groups

This test assumes that data follows a normal distribution. Since the data follows a normal distribution, we can proceed with this test.

Null Hypothesis : All the three error vectors are from populations with equal variances(homoscedasticity)

Alternate Hypothesis : The variances of the error vectors of the 3 models are not equal

Result:

```
BartlettResult(statistic=1.6061906634573557, pvalue=0.44794029222680476)
```

The p-value obtained from the Bartlett test on the three error vectors is 0.45 which is significant as it is greater than 0.05 which indicates that we fail to reject the null hypothesis. This indicates that the error vectors of three models satisfy the property of homoscedasticity required for the ANOVA test .

ANOVA test:

ANOVA test is used to compare the performance of different classifiers by comparing their error values. But ANOVA test assumes that the data should follow a normal distribution and equal variances among the data samples(homoscedasticity). To test those assumptions, NormalTest and BartlettTest are performed on the error vectors of the three models.The data is normally distributed and also satisfies the property of homoscedasticity. So ANOVA test can be performed on the error vectors.

Null Hypothesis : The means are equal for all the three error vectors indicating that all the three models have equal error values.

Alternate Hypothesis : At least one mean is significantly different in the three error vectors.

Result:

```
One-way ANOVA
=============
('F value:', 1682.3015743877436)
('P value:', 4.6019607889160175e-29, '\n')
```

The p-value of ANOVA test is not significant as it is much less than 0.05 indicating that we can reject the null hypothesis with more than 95% confidence. Hence, the means are not same for error vectors indicating that the 3 models have different error values.

Two-sample pairwise T-test:

The two-sample t-test determines whether the mean error values of any two models differ from each other in a significant way. It assumes that the error values are independent and normally distributed.

Null hypothesis: The mean error values of two models are the same.

Alternative hypothesis: The mean error values of two models are not equal.

We now perform paired t-tests on each pair of these models. The following are the results.

```
Doc2Vec-word2vectfidf
Ttest_indResult(statistic=-41.097213457305536, pvalue=3.004279365751239e-19)
word2vectfidf-word2vec
Ttest_indResult(statistic=-10.674215346865296, pvalue=3.2422843207047913e-09)
word2vec-doc2vec
Ttest_indResult(statistic=51.42392449908445, pvalue=5.4898923762105201e-21)
```

Models	P-value
Doc2Vec - Word2Vec	3.004279365751239e – 19
Word2Vec - Word2Vec + TF-IDF	3.2422843207047913e – 09
Doc2Vec - Word2Vec + TF-IDF	5.4898923762105201e – 21

In each of the above cases, we fail to reject the null hypothesis as p-value is not significant and hence conclude that the error mean values of no two models are equal.Hence, the mean error values are different for three models.

The following are the calculated means of three methods. By this we can tell that Doc2Vec model has least error rate. Hence, Doc2vec model has the best accuracy when compared to other models as seen from the accuracy results of each model.

8.2 References:

[1] Liping Jing, Lixin Zhou, Michael K. Ng, Joshua Zhexue Huang. Ontology-based Distance Measure for Text Clustering.

[2] Quac Le, Tomas Mikolov. Google. Distributed Representations of Sentences and Documents.

[3] Kira Schacht. Similarity and Distance in data.
http://journocode.com/2016/03/10/similarity-and-distance-part-1/

[4] Michael Czerny. Modern Method for Sentiment Analysis.
https://districtdatalabs.silvrback.com/modern-methods-for-sentiment-analysis

[5] Adrian Sanborn, Jacek Skryzalin. Stanford University. Deep Learning for Semantic Similarity.

Source: https://www.kaggle.com/c/quora-question-pairs

Dataset: https://www.kaggle.com/c/quora-question-pairs/data