**INTRODUCTION**

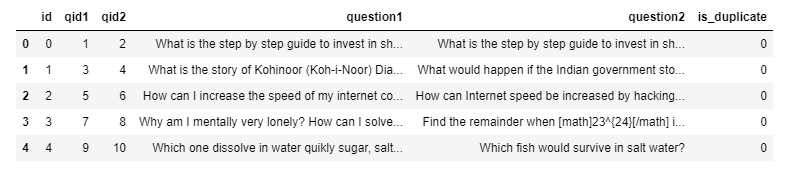
Machine learning is the scientific study of algorithms used for predicting and analyzing the data. It is a field which aims at improving the efficiency thereby making the machine learn by itself without much human intervention. It is a branch of artificial intelligence involving numerous statistical models used by the computer machines to perform the tasks and make decisions on the basis of the data and the recognized patterns. The ability to make the machine learn by itself is achieved through various algorithms some of which are performed in the present assignment:

* Logistic Regression
* XG Boost
* Random Forest
* Decision Tree

All the above algorithms are performed on a real-world dataset which can be downloaded from this [link](https://www.kaggle.com/c/quora-question-pairs/data). The dataset used in the assignment is named as **Quora Question Pairs** which aims at identifying the question pairs having the same intent. As we all know, Quora is a platform where a user post certain question and others’ try to contribute by replying to the asked question and giving useful insights. There can be instances where similar questions having the same intent been asked by different users in different versions. This results in seekers spending more time in searching the best answers to the questions with the same intent. Thus, there arises a need club this duplication so that writers don’t spend time answering multiple version of a question. Thus, in the present assignment, we have tried to classify the questions pairs into duplicate and non-duplicate with the help of the aforesaid algorithms.

**EXPLORATORY DATA ANALYSIS**

Firstly, the necessary packages are imported and then the dataset is extracted. After the extraction of the dataset, we have printed the data head to have a look at the data. The structure of the dataset is as follows:



We are given a minimal number of data fields here, consisting of:

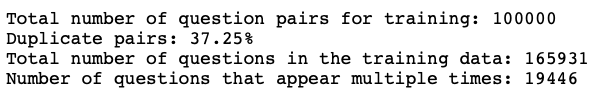
**id –** it looks like a simple row ID

**qid{1, 2}**: This represents the unique ID of each question in the pair

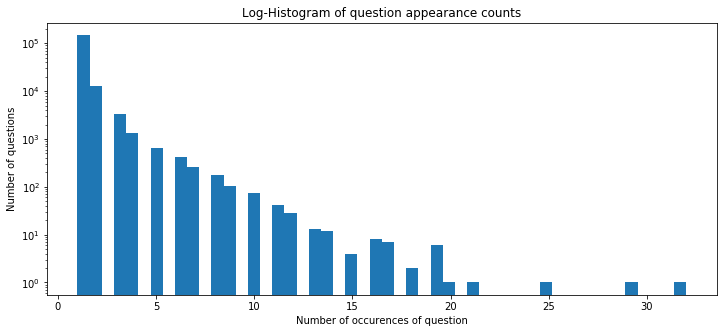
**question{1, 2}**: these columns contains the actual text content of the question in the pair.

**is\_duplicate**: This is the label that we are trying to predict i.e. whether the two questions are duplicates of each other.

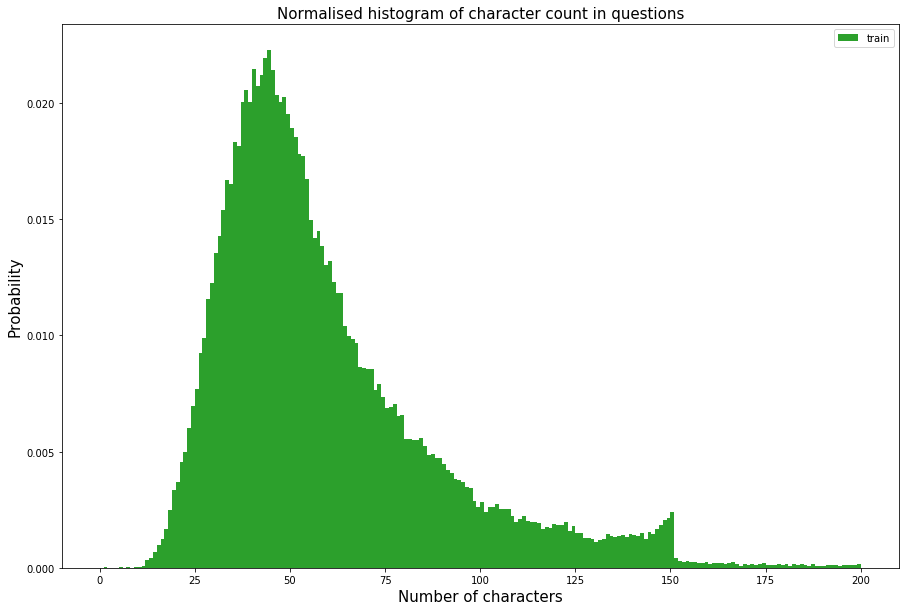
This is our dataset; we have explored and tried to find out certain things about the dataset such as the total number of questions present, the percentage of duplicate pairs, number of questions appearing multiple times, etc. The following image shows the result of the findings:



Now we have plotted the distribution of the question graph. The graph is as follows:



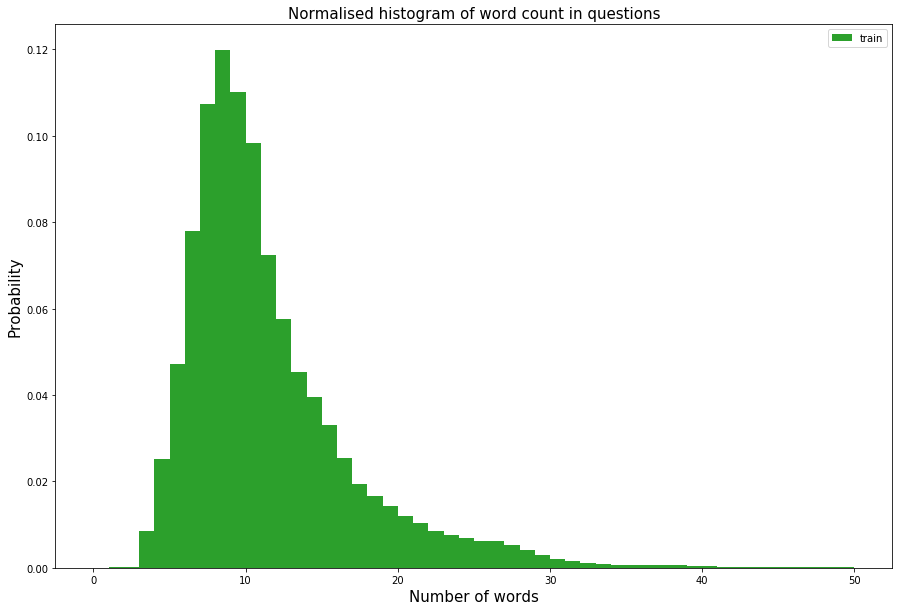
It can be seen from the graph that most of the questions are occurring 2-3 times. However, there are some questions which are occurring 25 times or 29 times even. Probably these are the questions which are related to something which is very trending or something which is related or concerned with a larger number of users. Now we have plotted another graph representing the character counts. The graph is as follows:



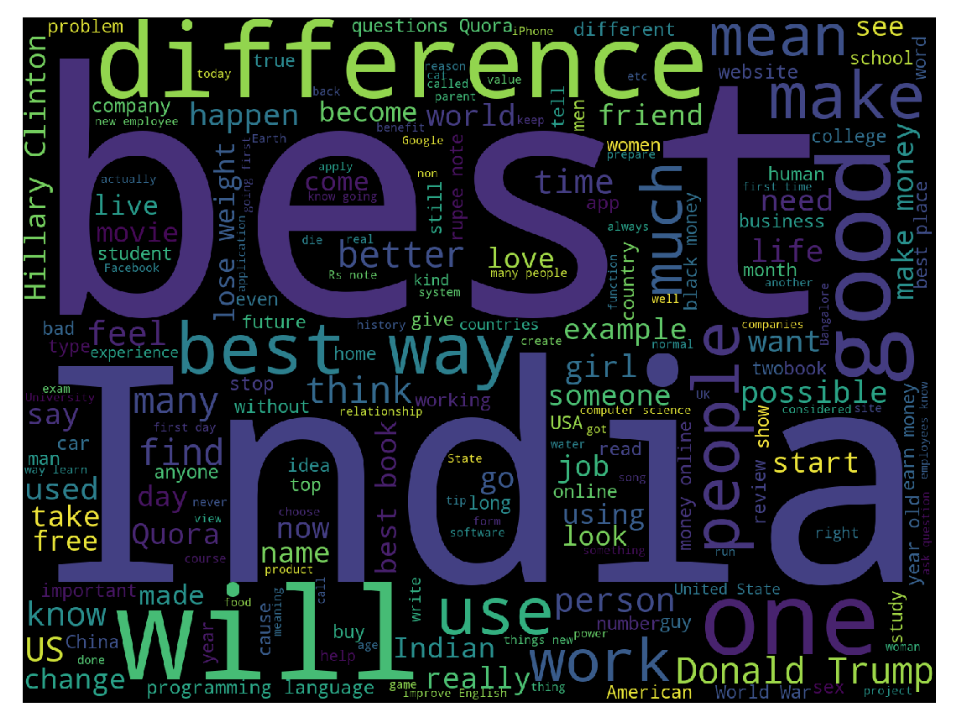
It can be seen that most questions have 15 to 150 characters in them. One thing that catches our eyes is the steep cut-off at 150 characters for the training set, for most questions. This could be some sort of Quora question size limit.

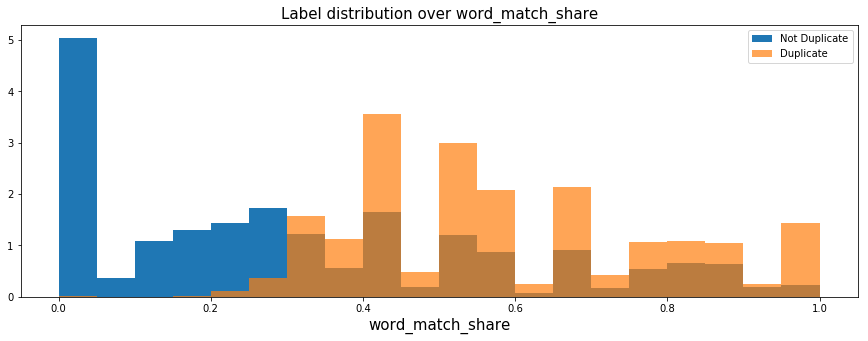
It's also worth noting that we have truncated this histogram at 200 characters, however, the entire data distribution is under 1200 characters. We have truncated at 200 characters because the samples with over 200 characters are very rare.

Similarly, we have also plotted the word count. In order to split the words, we have used a naive method i.e. splitting on spaces instead of using a serious tokenizer. This method would give us a good idea of the distribution.



We see a similar distribution for word count as that of character count. It can be seen that most questions are about 10 words long. It looks that the distribution of the data is pointy. After this, we have created a word cloud. A word cloud reflects the most common words used. It can be observed from the word cloud image that the words such as "best", "india” are most commonly used. However, the words such as “difference”, “will”, “way”, “one”, “Donald Trump” are not far behind in terms of commonality. The word cloud is as follows:





The above graph highlights the word match share in terms of duplicate and non-duplicate. We can see that this feature has a lot of predictive power, as it is good at separating the duplicate questions from the non-duplicate ones. Interestingly, it seems very good at identifying questions which are definitely different but is not so great at finding questions which are definitely duplicates.

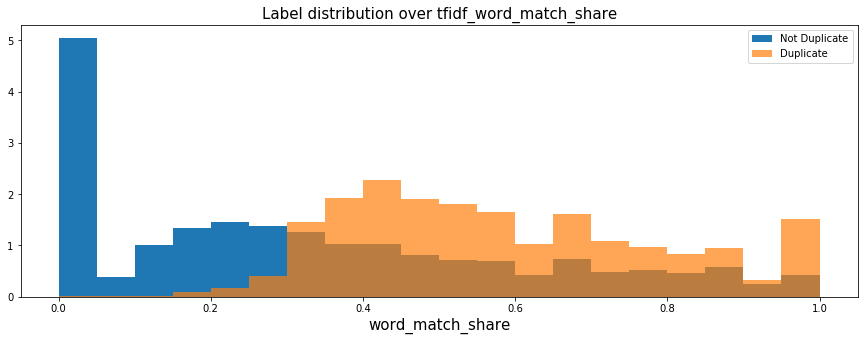
**DATA PROCESSING AND FEATURE ENGINEERING**

The first step of Data Processing is Data Cleansing. In order to increase the overall performance of the model, data cleansing is required so that the overall quality of the data can be improved. It is one of the necessary steps to be performed before fitting the model. The present dataset contains the data in text format and is gathered by the responses from the users on Quora, a lot of noise can be expected in the dataset. Therefore, techniques like Stemming, Stop Word Removal, Spell Check, Lemmatization, etc. have been used.

Stemming is a technique where words of the same root are clubbed to together. In better words, it tries to produce the morphological variants of a base word and clubs them together. For example words such as likes, liking, likely, liked are all clubbed to the same root word "like". Stop word removal technique has been used to remove the useless words in the data. Commonly used stop words include “a”, “an”, “the”, “in” etc. Spell Check, as the words suggest, is a technique which is used to spell check in a text. Since the data involves the responses given by the users, there may be many spelling errors which can be corrected with the help of this technique. Lemmatization is a technique similar to stemming but the only difference is lemmatization takes into consideration the meaning of words and the intended parts of speech in the text. For example, the word “good” can be considered as a lemma for the word “better”.

In the present dataset, all the above-mentioned techniques have been applied to get cleansed data. Once the data is cleansed, the questions are processed into vectors so that the data can be fitted into the machine learning model. The data is processed using word2vec and glove methods to reduce the dimensionality of the data. Then, the data is processed as question pair TF-IDF encodings.

TF-IDF stands for Term Frequency – Inverse Document Frequency. It is a numeric statistic which reflects the significance of a word in the document. The technique tells us how much important a word is in the document. By this technique, we weigh the terms on the basis of how uncommon they are, meaning thereby we care more about rare words existing in both questions than the common words. This makes sense, for example, we care more about whether the word "exercise" appears in both the questions than the word "and", as uncommon words will be more indicative of the content.



So, it looks like our TF-IDF actually got worse in terms of overall AUC, which is a bit disappointing. In the present assignment, we have used the AUC metric as it is unaffected by scaling implying it to be a good metric for testing the predictive power of individual features.

Now we have combined these features so as to make the predictions. But before fitting the model, some manual feature engineering is done like counting the number of important words in the question, ngrams and their embedding in lower dimensional spaces, etc. We know that the meaning of the word is determined by the words surrounded by it. An n-gram model is a type of probabilistic language model used for predicting the next item in such a sequence in the form of an (n-1) order Markov model.

**MODELING**

In the present assignment, we have tried to fit in 4 models:

* XG Boost
* Logistic Regression
* Random Forest
* Decision Tree

The first algorithm which we have performed is XG Boost. But before performing this, rebalancing of the data has been done as we have 37% positive class in our training data and only 17% in the test data. By re-balancing the data, such that our training set has 17% positives, we can ensure that XGBoost output probabilities will better match the data on the leaderboard and give us a better score (since LogLoss looks at the probabilities themselves and not just the order of the predictions like AUC).

After rebalancing the data, the model is fit, and predictions are made through XG Boost. The classification report is as follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1-score | Support |
| 0.0 | 0.99 | 0.82 | 0.89 | 37915 |
| 1.0 | 0.09 | 0.59 | 0.15 | 1105 |

It can be seen from the classification report that the model is good but not great. The precision and recall are higher for the one class but not for the other. The ROC score is also 0.705 which is average.

The next model which is used for making the prediction is Random Forest. Random Forest is an algorithm which often gives accurate results since it merges the multiple decision trees. It can be seen that the ROC score is 0.796 which is quite good. The classification report also highlights the same.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1-score | Support |
| 0.0 | 0.95 | 0.89 | 0.92 | 33408 |
| 1.0 | 0.52 | 0.70 | 0.60 | 5612 |

It can be seen that the precision and recall is higher than the previous XG Boost. Both the classes are recalled and also the F1 – score is improved.

After Random Forest, the next model which we have fitted is the Decision Tree. Although the results of the decision tree may not be as great as Random Forest, they are still better or at par when compared to other algorithms. However, in the present assignment, it can be seen that the decision trees model does fit well and produces similar results as that of the random forest. The classification report of the decision tree is as follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1-score | Support |
| 0.0 | 0.95 | 0.89 | 0.92 | 33449 |
| 1.0 | 0.52 | 0.71 | 0.60 | 5571 |

The ROC score is 0.798 which is pretty good. The classification of the decision tree is very similar to that of the random forest. This highlights that both decision tree and random forest are pretty good on the current dataset.

Now, the last model which we have fit in order to make the prediction is that of Random Forest. The ROC score of Random Forest is 0.608 which is not good. Even the precision and recall have been reduced resulting in the decrease in F1 score.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1-score | Support |
| 0.0 | 0.95 | 0.82 | 0.88 | 36506 |
| 1.0 | 0.13 | 0.40 | 0.20 | 2514 |

**EVALUATION & ANALYSIS**

In the present assignment, predictions have been made through 4 models and their evaluation is done on the basis of ROC score. ROC stands for Receiver Operating Characteristics. It is a measure of separability. ROC score represents the performance measurement of the classification problem. In order to check the performance, ROC score is analyzed, and the conclusions are drawn. The ROC represents the capability of the model in distinguishing the classes. Higher the ROC score, better the model in making the predictions. Higher the score, higher chances of predicting 0s as 0s and 1s as 1s. The following table shows the comparison of the ROC scores:

|  |  |
| --- | --- |
| **Models** | **ROC score** |
| Logistic Regression | 0.60 |
| XG Boost | 0.70 |
| Random Forest | 0.79 |
| Decision Tree | 0.79 |

From the above table, it can be analyzed that Random Forest and Decision Trees models have made the most accurate predictions as compared to the other models. It can further be analyzed that the ROC score of both Random Forest and Decision Tree is similar. This is because of the fact that Random Forest merges the multiple decision trees so as to give accurate results. The core of Random Forest is a Decision Tree.

It has been observed that Logistic Regression is underfitting the data even after feature engineering. Thus, it is because of under-fitting the Logistic Model is not predicting well. Also, the XG Boost underfits the data due to the bad hyperparameter tuning and it is extremely difficult to tune the hyperparameters as the data size is very large and XG Boost runs sequentially.

**CONCLUSION**

This brings an end to the report. On the basis of the analysis, it can be concluded that machine learning is the study of algorithms used for predicting and analyzing the data. It is a field in which machines are trained to respond on the basis of the data and the recognized patterns. It has been done by with the help of various algorithms which are nothing, but a set of rules used in training the computer to work with less human intervention.

In the present assignment, predictions about the duplicate questions have been made on Quora Question Pairs dataset and it has been found that Random Forest and Decision Tree predict well on the dataset. We have done some feature engineering but there is a scope to add a lot of features to boost the accuracy and also tune the hyperparameters for better modeling.

Thus, it can be concluded that the success of predictive analysis depends upon various things i.e. ability to mine the data, the technique or modeling used, the prediction which is to be made and lastly intelligence to generate and analyze the results. The entire process should be viewed as the amalgamation of these rather than the discrete and indivisible process.

**REFERENCES**

* Sarang Narkhede. (2018, June 26). Understanding AUC - ROC Curve. Retrieved from <https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5>
* Salman Khan. (2016, October 13). Predictive analytics – knowledge is power. Retrieved from <https://www.itworldcanada.com/blog/predictive-analytics-knowledge-is-power/387243>