QUORA QUESTION PAIRS

Aman Arya, Navneet P. Singh

DALHOUSIE UNIVERSITY

Problem Statement

Quora question pairs is one of the Kaggle competitions where the objective is to predict whether the given question pairs have the same meaning. Our main objective is to create a model and train the model to predict whether the question pairs are similar or not.

Approach to solve the problem

For this purpose, our first step was to understand the dataset provided by Quora [1]. We were given a training data set and a test data set. Training data set contains three columns. First two columns contain the question text and the third column shows whether the sentences are similar or not. We used the training data set to train our model and test data set to predict whether the given question pair is similar or not.

Once the data set was figured out, we approached this problem with three possible solutions. Initially, we started out by the data cleaning process (e.g. removing stop words, stemming, lemmatization, etc.) to obtain the clean data set. Further, we created the tfidf for the given sentences and used this tfidf matrix to create the model ahead.

Once the data cleaning part was completed, we went ahead to approach this problem with three possible solutions. First, was to use the cosine similarity [2] to find out whether the question pairs were similar or not. Further, we used confusion matrix to find out the accuracy of the result obtained against the given output.

Second approach was to use a machine learning algorithm (i.e. Random Forest) [3] to find out the similarity of the question pair. We would train our model using the training data set provided and test our accuracy using the test data set against the given output. Further, we would use a confusion matrix to find out the accuracy of the result obtained against the given output.

Using the above algorithm, we would find out which algorithm gives a better accuracy and use that algorithm to predict whether the two questions are similar from our test data set.

The entire process would be implemented using python with NLTK library.

Such type of challenge has a lot of applications, for example, we can extend it to create a plagiarism software [4] that detects if two submission are similar or not.

Project Plan

Task	Status
Data cleaning (stemming, removing stop words, etc)	Completed
Creating tfidf from training data set provided	Completed
Using cosine similarity to find out if the questions are similar or not	Completed
Creating confusion matrix from the results obtained from cosine similarities	Completed
Using machine learning model to find out if the questions are similar or not	Pending, would be implemented within 11th - 17th March
Deciding which algorithm to use to predict the similarity	Pending, would be implemented within 11th - 17th March
Applying the logic for test data set	Pending, would be implemented within 18th - 24th March
Finishing and creating presentation	Pending, would be implemented within 25th - 31st March

References

[1] Kaggle.com. (2019). *Quora Question Pairs* | *Kaggle*. [online] Available at: https://www.kaggle.com/c/quora-question-pairs/ [Accessed 11 Mar. 2019].

[2] http://dataaspirant.com/2015/04/11/five-most-popular-similarity-measures-implementation-in-python/

[3] Towards Data Science. (2019). *Natural Language Processing on multiple columns in python*. [online] Available at: https://towardsdatascience.com/natural-language-processing-on-multiple-columns-in-python-554043e05308 [Accessed 11 Mar. 2019].

[4] PyPI. (2019). *pycode-similar*. [online] Available at: https://pypi.org/project/pycode-similar/ [Accessed 11 Mar. 2019].