Machine Learning Engineer Nanodegree

Capstone Proposal

Rishab Ghanti

27th April 2018

Proposal

Quora Question Pair (Kaggle Competition)

Domain Background

Quora is a very famous platform (website and Android app) all over the world where questions can asked by anyone in the community which are answered by Quora's 100 million monthly users. There are different platforms like *StackOverlFlow, Yahoo! Answers, WikiAnswers* etc. for different domains but Quora is by far the most popular general question-answer platform.

Due to its huge user base, it is expected that many users ask questions structured differently but having the same meaning. For example, the questions 'What is the population of USA' and 'how many people live in USA' are duplicates. Marking both these as duplicates and showing users one version of the question with the correct answer will allow users to get answers to their questions quickly and giving them more value.

A very useful resource that I found which is solving the problem in a similar way that I intend to. I will use this link for further reference.

Problem Statement

The goal of this project is to identify from the given question pairs, which pair is a duplicate, i.e. which pairs have questions with the same meaning. This is a natural language processing problem which is a popular field in Artificial Intelligence. I plan to use advanced techniques in machine learning that I learnt during the Nanodegree course to solve this problem. Solving this problem will allow users of Quora to have a more valuable experience on the platform.

Datasets and Inputs

The dataset that I plan to use in this project has been provided by Quora as a Kaggle competition.

The input fields are:

1. id – id of the training set question pair.
2. qid1, qid2 – unique ids of each question (available in train.csv).
3. question1, quesion2 – full text of the question
4. is_duplicate – set to 1 if the questions have the same meaning, else 0.

The total number of entries in this data set is 4,04,290 with 1,49,263 being positive entries (is_duplicate = 1) and 2,55,027 of them being negative entries (is_duplicate = 0)

Hence, the training set has 363,861 (90%) entries and the testing set has 40,429 (10%) entries.

Solution Statement

The solution to solve this problem will be prediction if the question pairs are duplicates or not. I plan to extract some features such as word length, similar words between the questions in the pairs etc. and do some visualization to get a better understanding of the data. Then I plan to do feature extraction and perform some function on the features such as max pooling to decrease the computation power and time I will need to train my model. I will then build a deep neural network to train the data and find the accuracy for the model based on the evaluation metric explained below. I will tweak different parameters of the neural network to get the best accuracy model which I will use on the testing dataset. I plan to 90% of the dataset as training data and 10% as testing data. To improve the training of the model I will use dropout layers to better train the neutrons in the neural network.

Benchmark Model

The benchmark model that I have considered is to find the percentage prediction of an event, i.e. x percent prediction that the pair is duplicate and (100 – x) percent prediction that it is not. A baseline model to do this is to look at the bag of words similarity of duplicate and non-duplicate questions using the Jaccard index. This is a naïve method and hence I plan to achieve a better model than this in my capstone project.

Evaluation Metrics

I am going to use prediction accuracy as the evaluation metric for this project. The equation I intend to use is:

$$aaccuracy = \frac{true\ positives + true\ negative}{total\ samples}$$

True positive – the number of pairs that contain duplicate questions and the model has detected them correctly

True negative – the number of pairs whose questions do not mean the same and it has been correctly identified by the model as non-duplicates

Total sample – total number of pairs in the dataset

After further exploring the data set I see that the dataset is unbalanced, i.e. it has 36.91% of positive entries whereas it has 63.08% of negative entries. Thus, *accuracy* will not be the best metric to judge this model since there will be a lot of true negatives compared to true positives.

I plan to F-score as the main evaluation metric for my model in addition to *accuracy*.

$$F = 2 * \left(\frac{precision * recall}{precision + recall}\right)$$

precision – number of positive results divided by the number of positive results returned by the classifier

recall – number of correct positive results divided by the number of all relevant samples (all relevant samples that should have been identified as positive)

Project Design

First I plan to explore the data and see how it is formatted and how I can use each of the columns to improve my model. Then I will extract different features such as number of positive entries and negative entries in the data set, word count of the questions, number of common words in the question pair etc. I plan to use the Jaccard index to see if the number of common words in the questions can help me better predict if the questions are duplicates or not.

I plan to the do the following to do build a good model:

1. Feature selection - Since the Quora dataset does not have a lot of features, this step will be pretty easy.
2. Feature transformation – This step would include tokenizing the questions and finding word vectors for them which would give me a matrix for each question. To reduce the size of the matrix I plan to use a max pooling layer.
3. Neural networks – I then plan to create a neural network probably a deep neural network. I plan to use Siamese Manhattan LSTM deep neural network to solve this problem similar to this article (very useful link!).

To better train the model and improve the accuracy of the model, I will use dropout layers. I am not 100% sure about the model and it may change once I start coding and tweaking the parameters to improve the performance of the model

References

- Kaggle
- Quora