Machine Learning Engineer Nanodegree

Capstone Proposal

Andre Mechali
April 27th, 2017

Proposal

Domain Background

The problem I will try to solve is to determine, given 2 different questions, if their true meaning is basically the same. This is part of a Kaggle competition hosted by Quora. They need this information because it's common that users formulate questions that have already been answered, and therefor Quora can avoid repeating them. The problem can clearly be solved, because they already have their own machine learning algorithm for doing it. Other platforms, such as StackOverflow, have also their own techniques for doing this. However, this information is not public and therefor cannot be used as a benchmark.

I find this subject very interesting because it combines machine learning with natural language processing, which is a quite challenging and important subject nowdays. It's also a very important field because it can be applied for a wide range of problems, such as translators or personal assistants like Siri.

Problem Statement

The problem to be resolved is to determine if 2 different questions, made sepparately by Quora users, refer to the same question, despite the fact of not being written in the same way. The results can be evaluated on the log loss between the predicted values and the true values of the testing dataset.

Datasets and Inputs

The dataset is provided by Quora for its Kaggle competition. It consists on a training and testing dataset, where the first has 2 questions, the ID of each question, and a label stating if both questions have the same meaning. This label was human-made, so it is subjective but accurate in most cases. The testing consists only on pairs of questions, some of which are generated by computer instead of humans. These datasets can be obtained from the following link: https://www.kaggle.com/c/quora-question-pairs/data (https://www.kaggle.com/c/quora-question-pairs/data) .

From the training dataset, which has about 400,000 pair of questions, it is possible to identify patterns that appear in questions with the same meaning, and apply them on the test dataset. The questions used for training correspond to Quora real questions, and will be applied for the same use, so they are completely relevant.

Solution Statement

In this section, clearly describe a solution to the problem. The solution should be applicable to the project domain and appropriate for the dataset(s) or input(s) given. Additionally, describe the solution thoroughly such that it is clear that the solution is quantifiable (the solution can be expressed in mathematical or logical terms) , measurable (the solution can be measured by some metric and clearly observed), and replicable (the solution can be reproduced and occurs more than once).

A solution for this problem should state for any given pair of questions, if the meaning is the same or not. This will be represented as a 1, if they mean the same, and a 0 if not. This can then be easily compared to the label that each question has stating weather this is true or not, which is also represented by a 1 or a 0. For any given pair, the algorithm will be the same and therefor generate the same result if it is applied several times.

Benchmark Model

The benchmark to consider in this case will be applying the probability for two given questions of being considered the same. For this, I will measure the proportion of pairs in the training set that refer to the same question, and assume that this is the probability of a new pair of having the same meaning. So for every new pair, the result will be the same. This process will then have a simple output, which will be 1 with some ***p*** probability and 0 with probability *1 - p*.

This model is easily measurable, using the same criteria than the final model that I will develop for this project. This metric, as stated before, will be the log loss between the predicted and the true values.

Evaluation Metrics

(approx. 1-2 paragraphs)

In this section, propose at least one evaluation metric that can be used to quantify the performance of both the benchmark model and the solution model. The evaluation metric(s) you propose should be appropriate given the context of the data, the problem statement, and the intended solution. Describe how the evaluation metric(s) are derived and provide an example of their mathematical representations (if applicable). Complex evaluation metrics should be clearly defined and quantifiable (can be expressed in mathematical or logical terms).

The evaluation metric used for measuring the performance of both models will be the log los beween the predicted and the true values. The main reason for choosing this is that the Kaggle competition which provides this data uses that metric. For this reason, and because I don't have the true values for the testing set, I cannot use another metric. While developing the model, however, the training dataset will be split so that I can test it by my own.

The mathematical formula for determining the log loss is the following:

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

where y is the real value and p the predicted one.

Project Design

For solving this problem I will apply, on the first place, some transformations for preprocessing the data. First of all, indentify and analyze outliers, and determine if they should be excluded. After that, some transformations to each question, trying to obtain the important words. For example, I will remove the stop words, which are likely not relevant for the meaning of the question, and use a stemmer to obtain the root of each word. I will also apply a TF-IDF transformation to determine the relative importance of each word in the dataset. Finally, I will experiment with synomyms, to see if it is possible to consider different synonyms and if they were the same word. With all this new data, I will create a series of new features, like the length of the questions, the shared words, etc. The process described before will consist on a series of opperations that will be completely replicable, and will generate the same results if applied several times for a given dataset.

Having already preprocessed the dataset, I will be able to try different models to find a relationship between these new features and the label of each pair of questions. I will try 3 algorithms for this problem: Random forest, Adaboost and XGBoost. As said before, they will all be evaluated using log loss. Despite having a training and testing set already provided by Kaggle, they are too big for experimentation. Because of this, I will start with a random subset taken from the training set, as well as a testing set. With this, I will be able to compare the 3 models faster, and determine which might be the best one for this problem. Once this is done, I will try this algorithm with the whole set, and find the parameters which allow me to get the best performance.