Machine Learning Engineer Nanodegree

Capstone Proposal

Kamal Kaushik

(kmlkshk@gmail.com)

February 28th, 2018

Quora Question Pairs (Kaggle)

Domain Background

Quora is basically a website where people can ask any kind of questions and they are answered or edited by its community members who have knowledge in those specific areas.
Since millions of people use Quora to get answers of their queries, so you will find many people asking similar questions. Multiple answers to one question can lead in spending a lot of time in finding the best answer to the question raised by a member. Here, our aim is to find an algorithm and to apply that so as to gather all duplicate questions together and list them out when a member is looking for an answer to that question. Also, it will help members to answer the questions all at once who wish to do so.

Random Forest model is being currently used in Quora which identifies all duplicate questions. But, in present scenario there are many new algorithms which are available and are showing better outputs then Random Forest Model. To tackle this natural language processing problem I will use Natural Language Sentence Matching (NLSM), as a basic task in NLP.   This will be integrated into many NLP tasks to measure the degree of match / similarity / task relevance between two natural sentences. Basically, NLSM is the task of comparing two sentences and identifying the relationship between them.

Problem Statement

This is a binary classification problem where I will be using the Quora dataset from Kaggle competition and this is related to the problem of identifying duplicate sentences. Inputs are two sentences of texts and the goal is to predict if these two sentences are of the same meaning or not.

I will be tackling this as a natural language processing problem and will use both TF-IDF vectorization and word2vec (To check which one is giving the better output) to process input texts and will finally implement the one which is better .

Then I will use techniques such as regression and decision trees to train the dataset. The features will be extracted from sentences such as word count, character count and word distribution.

The duplicate detection problem will be defined as follows: for example Q1 and Q2 is a set of questions , the model will learn the below function :

f(Q1, Q2) → 0 or 1

where 1 represents that Q1 and Q2 have the same intent and 0 otherwise.

Datasets and Inputs

The Quora dataset is a set of questions , with annotations which indicates whether the questions seeks the same information or not. This data set is large, real, and relevant — a rare combination. Every line has a unique ID for each question in the set and a binary value which is 0 or 1 indicates whether the line has a duplicate pair or not. We will consider that the dataset is divided into two files, training and test, below is a quick analysis-

Datasets

1. **Training**:

   - Question pairs: 404290
   - Questions: 537933
   - Duplicate pairs: 36.92%

2. **Test:**

   - Question pairs: 2345796
   - Questions: 4363832
   - Question pairs (Training) / Question pairs (Test): 17.0%

Input Data fields

- 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.

Solution Statement

The solution will be whether the set of questions are duplicate or not in the test dataset.

- First I will try TF-IDF and word2vec both to see the better result to process all the texts and do some visualization of the data to get some understanding.
- Then I will proceed with the one algorithm which shows better outputs.
- Finally, I will perform feature extraction and select features such as word length, word count distribution and character count.

I will use the following models :

Linear Regression (as a base model) and
XGBoost (as a trusted algorithm) and if required, Deep Learning.

Benchmark Model

At present, the benchmark model is random forest model but, since I will try other algorithms also. Hope will get a better one with better outputs.

Evaluation Metrics

Prediction results are evaluated on the log loss between the predicted values and the ground truth. According to Kaggle competition webpage, the ground truth is the set of labels that have been supplied by human experts. The ground truth labels are inherently subjective, as the true meaning of sentences can never be known with certainty. Human labeling is also a 'noisy' process, and reasonable people will disagree. As a result, the ground truth labels on this dataset should be taken to be 'informed' but not 100% accurate, and may include incorrect labeling1.

Project Design

In the initial phase of the project I'll explore the data and will do a detailed Data Analysis, both on training as well as on testing dataset. Then I will start doing my natural language processing and extract information such as character counts, sentence length, TF-IDF, word2vec etc..

For better understanding of the data distribution I might perform some graph visualization also.

To train models, I will use some different models to compare. Since, this is a classification problem, I'll also use different approaches to get better results. The model which gives better results will use that to get the final outputs.

The final accuracy will be calculated against the test data set provided by Kaggle, by using the log loss between the predicted values and the ground truth.

References

1. https://www.kaggle.com/c/quora-question-pairs
2. https://www.quora.com
3. https://deeplearning4j.org/word2vec
4. https://zhuanlan.zhihu.com/p/31509849
5. https://engineering.quora.com/Semantic-Question-Matching-with-Deep-Learning
6. https://deeplearning4j.org/bagofwords-tf-idf