Duplicate Question Detection

bjk224, ag983, ys525

February 26 2018

1 Research Context

People use online question answering platforms such as Quora and Piazza to get there questions answered and to explore the collective knowledge of their peers. Quora is a more broad question-answering service, while Piazza is subdivided into different classes at different universities. The fundamental problem that we will look at is identifying if your question has already been answered. Duplicate questions cause seekers to waste time looking at multiple different threads to find their answer, and results in writers spending extra time writing similar answers. Having just one of each question makes for a much clearer experience, and results in more collaboration towards a better answer.

2 Research Objective

The objective of the project is to build a model that can predict whether or not two questions are duplicates. Duplicate questions are questions where one answer would be sufficient to both. However, we will be identifying duplicate questions from the questions alone, so we will look for questions about similar topics with similar questions words. The dataset that we will be using is from a Kaggle Competition on a dataset given by Quora. We will train the classifier on that dataset, and then our goal is to see how we can implement it into a setting like Piazza to improve that learning experience.

3 Current Methods

Currently, Quora uses a random forest classifier to identify duplicate questions. Our goal is to use a high dimensional representation of duplicate questions based on the context, and then sort most similar questions via KNN (K-nearest neighbors). However, we will first start our exploration of the dataset by looking into some of the top performers of the challenge and seeing if we can replicate their results. We will use Quora as our case study, and then adapt our model to work on Piazza. Piazza is a slightly different challenge because questions about a certain class have very similar context, but might be slightly different and require a slightly different answer.

4 Timeline

We are not yet proposing a specific timeline for each of the events that need to occur, rather I will just list what we need to do in the order that I see it getting completed.

- Download Quora dataset
- extract features based on the first place solution of the kaggle competition
- Word2Vec encoding of question sentences
- Train/Test with random forests classifier on the dataset
- Train/Test with KNN classifier
- Train/Test other classifiers based on how well the previous ones did
- Train/Test a GRU-based sentenced encoding technique
- At the same time as the above, start scraping data from Piazza so that we have it ready when needed.
- Read some of the other winning solutions to the problem and try to adapt their solutions.
- Deploy our method on piazza data, and develop method to get rid of duplicates (either based on time question was posted or if one of the duplicates has already been answered)

5 Division of Work

Brandon - Word2Vec encoding, random forests/KNN
Arnav - LSTM based classification
Zhao - GRU-based encoding

6 Resources

We will use the Quora dataset available from the kaggle competition. The testing dataset has is roughly 314MB, and has 2.3 million pairs of questions. The training dataset is smaller, approximately 60MB with 400,000 pairs of questions. Validation will be pretty simple for this because we will exclusively be using supervised learning models, and therefore we will immediately have benchmarks for how well we are doing. After we feel confident on the Quora dataset, we will move on to using data from Piazza. We will only be able to access data from open classes at Cornell on Piazza. We should be able to find an unoffical API in order to extract the question data from Piazza, but it could prove to be difficult. Additionally, validation will be tricky because we will either have to manually label the data, or trust that our well-performing algorithm on Quora data is directly transferable. The Piazza dataset will just be a dump of questions without any sorting, and therefore we will have to adapt the algorithm to pit each question against every other question and look for pairs. This would be very costly so we will have to develop another method in order to bin the questions before running our model.