Identifying Duplicate Questions

Yushi Homma, yushi@stanford.edu
Stuart Sy, stuartsy@stanford.edu
Christopher Yeh, chrisyeh@stanford.edu
February 9, 2017
Mentor: Kevin Clark

1 Problem Description

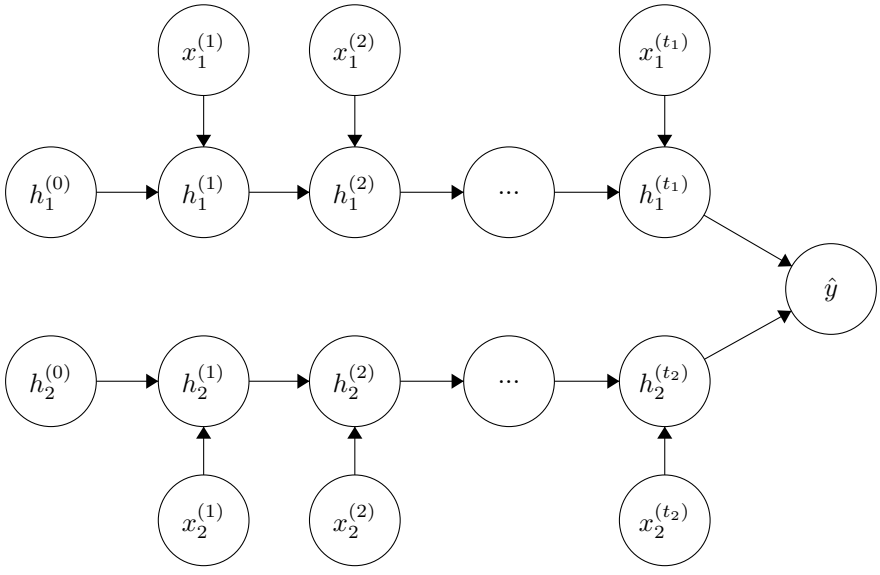
Detecting duplicate questions has become a prevalent and difficult challenge for online Q&A forums, which rely on this to keep answers for the same question in the same place. To identify duplicate pairs of questions, we use the definition of semantically equivalent questions proposed by Bogdanova et al. [1], that two questions are semantically equivalent (duplicates) if they can be answered by the exact same answer. Our project will be to determine how to best leverage deep neural networks to identify these duplicate pairs of questions from non-duplicate pairs.

2 Data

The data we will use for this project the Question Pairs dataset released last month by Quora. The dataset consists of 400,000 potential pairs, with 149306 duplicate and 255045 non-duplicate pairs. Each pair of questions is labeled with either 0 or 1, indicating whether or not the pair is a duplicate, or semantically equivalent.

3 Methodology/Algorithm

The algorithms we plan on using are inspired by the recurrent neural network algorithm in the Sanborn-Skryzalin paper[3]. The general outline of the simplest of our algorithms is using a recurrent neural network to convert two inputed questions independently into two hidden vectors, from which we use some linear transformations and a nonlinear thresholding function to calculate a prediction for the equivalence of the pair.



The majority of the algorithms we will try will have this general layout; however, we also plan to try to incorporate features from the other question in the pair in each of the final hidden vectors before making a prediction. We also plan to try augmenting our dataset by creating more non-duplicate pairs of questions by combining different lines of the existing dataset.

4 Related Work

Detecting semantically equivalent sentences or questions has been a long-standing problem in natural language processing and understanding. In recent years, the application of deep learning to this problem have offered significant improvements over traditional machine learning methods. Bogdanova et al.[1] found that their convolutional neural network algorithm was more effective than the traditional methods of using Jaccard similarity or support vector machiness in determining whether two questions are duplicates. Sanborn and Skryzalin [3] compared the use of recurrent neural networks and recursive neural networks with traditional machine learning methods and found that recurrent neural networks performed the best on the SemEval-2015 dataset. Their recurrent neural network algorithm will serve as a model for our algorithms in this project. Dey et al. [2] studied alternative methods other than deep learning to approach this problem, arguing that deep learning has inherent flaws that make it unsuitable for detecting semantic similarity. We will use these results to help refine our model beyond the current prevaling models for semantic similarity detection.

5 Evaluation Plan

Since our dataset is entirely labeled and has enough data to split into a training set and a test set, we will evaluate the effectiveness of our models by testing its success (% correctly labeled) on a predesignated test set. We expect to have tables of results with the percentage of correctly labeled pairs for each algorithm we use. We also expect to create confusion matrices for the algorithms to analyze what flaws our models have.

6 References

[1] Bogdanova, D., Santos, C.D., Barbosa, L., & Zadrozny, B. (2015) Detecting semantically equivalent questions in online user forums. *Conference on Computational Language Learning.* **19**: 123-131.
[2] Dey, K., Shrivastava, R. & Kaushik, S. (2016) A paraphrase and semantic similarity detection system for user generated short-text content on microblogs. *International Conference on Computational Linguistics: Technical Papers*, **16**: 2880-2890.
[3] Sanborn, A. & Skryzalin, J. (2015) Deep learning for semantic similarity. *CS 224d: Deep Learning for Natural Language Processing.* Stanford, CA.