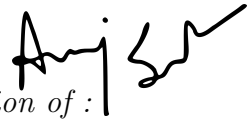
Atal Bihari Vajpayee Indian Institute of Information Technology and Management Gwalior

INFORMATION TECHNOLOGY

Mini Project Final Report

Quora Insincere Questions

Submitted by:
ANKIT MAURYA 2018IMT-018



Under the supervision of :
Dr. Anuraj Singh

ABSTRACT

Every day, thousands of questions are asked in QA forms, and manually classifying them as "sincere" or "insincere" is impractical. Insincere concerns have a negative impact on the platform's user experience and are a major source of concern. As a result, we introduce a machine learning model for Question classification to dilute such questions. Quora provided the dataset through the website Kaggle, which includes a training set of over 1.3 million labeled examples and a test set of about 300,000 unlabeled examples. A Long Short-Term Memory (LSTM) unit and a recently proposed gated recurrent unit were implemented as artificial recurrent neural network (RNN) architecture (GRU).

TABLE OF CONTENTS

LIST OF FIGURES

1 INTRODUCTION

Quora is a common online forum where users can ask questions and (usually) receive thoughtful responses. Although the majority of questions are asked in good faith, a small percentage of bad actors ask questions that are either insincere or problematic (for example, questions based on false premises or questions that are simply intended to make a statement). In order to improve the overall group experience, a challenge is to create a classifier that will take a user-generated query as input and automatically identify it as genuine or fake.

1.1 Motivation

The main goal of this project is to transform an NLP problem (predicting insincere questions) into a generalized machine learning problem that can be solved using a variety of machine learning techniques.

To make the internet a healthy place for information sharing, it is imperative that toxic and divisive content be filtered out. A recent attempt is the Quora challenge.

1.2 Literature Survey

In their paper "Exploring Deep Learning in Semantic Query Matching," Ashwin Dhakal and his co-authors used an Artificial Neural Network approach to predict semantic coincidence between question pairs, extracting highly dominant features, and evaluating the likelihood of a duplicate question on Quora. Words and phrases are mapped into real-number vectors in their research, then feature engineering is applied, which involves NLTK mathematics, fuzzy wuzzy features, and Word mover distances combined with vector distances.

Hence, the research has discussed and following the architecture used by the Quora itself along with the knowledge of natural language processing and machine learning.

According to a CMU (Carnegie Mellon University) scholar, the paper titled "Hierarchical Attention Networks for Documents Classification" was co-authored by CMU and Microsoft Research in 2016. Six large-scale Datasets were used in the experiment. The tests revealed that this model outperforms the previous best baseline approaches by 3.1 percent and 4.1 percent for smaller datasets like Yelp 2013 and IMDB, respectively. On Yelp 2014, Yelp 2015, Yahoo Answers, and Amazon Reviews, this model outperforms the previous best model by 3.2 percent, 3.4 percent, 4.6 percent, and 6.1 percent for large datasets.

1.3 Objectives

The main goal of this project is to determine if a Quora query is sincere or not. Insincere questions aren't intended to elicit useful information on a subject of interest; instead, they may be rhetorical or claims. People can tell them apart from normal, sincere questions because of this peculiarity:

1.The tone, which is exaggerated to attract attention to a point or a position rather than being non-neutral.

2.Their ultimate intent is to inflame, suggesting a discriminatory idea, seeking confirmation of a stereotype, insulting a person or a group, often basing upon characteristics that are not measurable

3.They are generally not based on evidence, nor have a solid connection with reality They can contain sexual content in order to be more offensive and elicit a wider response (disdain or shock) from the user community.

2 METHODOLOGY

2.1 System Architecture

1.Model consists of word embeddings, followed by BiDirectional LSTM layer, and a few dense layers to make final prediction.

2. GlobalMaxPooling layer helps to detect match at any time step instead of taking results from the last node.

3. To minimize overfitting, I used a dense layer followed by a dropout layer.

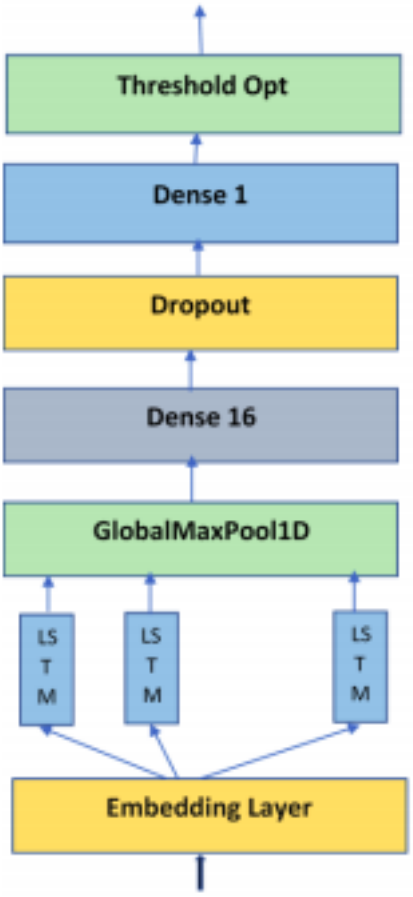


Figure 1: Model architecture

2.2 Block Diagram Techniques Used

Recurrent neural networks are a form of neural network in which the output is fed back into the inputs. Long short-term memories, or LSTMs, are an architectural version of recurrent neural networks (RNNs) that can be used to solve the vanishing gradient problem. They are working on a mechanism for using previous data in order to generate output.

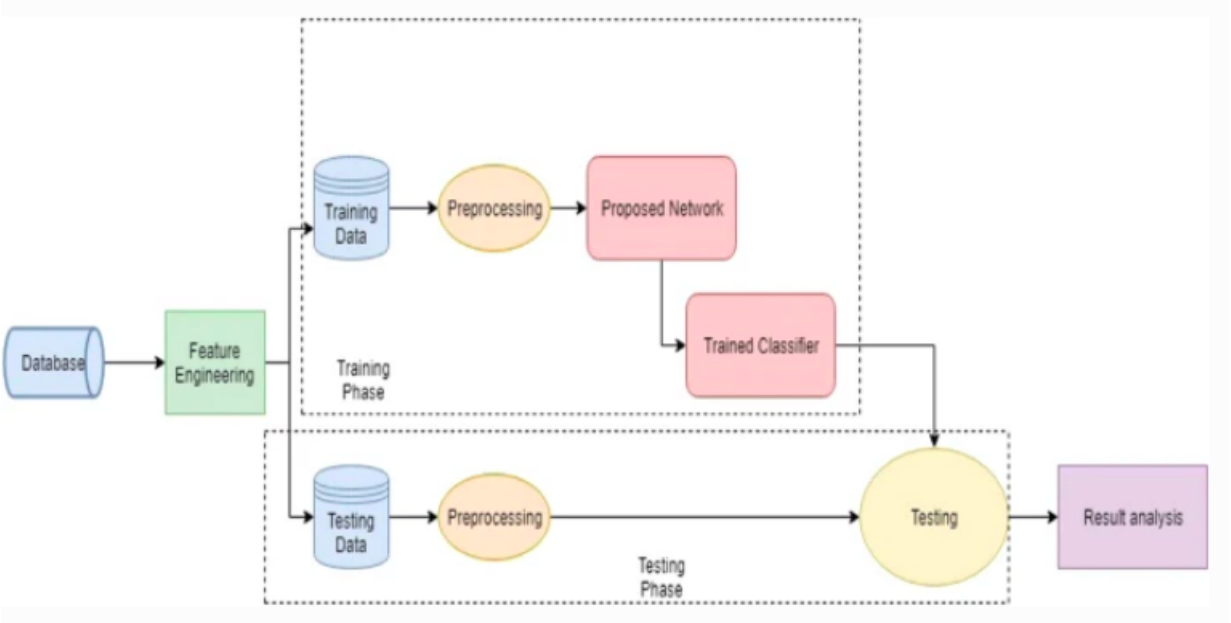


Figure 2: Proposed system

2.3 Methodology

The most important aspect of a good project is its implementation using the appropriate methods. Various methodologies, such as KDD and CRISP-DM, allow for efficient project growth. CRISP-DM, which stands for Cross-Industry Process For Data Mining, will be used in this project. It's a technique that's a more advanced variant of the KDD system. We may update the model and make adjustments to previous stages at any point of the project using this method. This gives the project room for change. The CRISP-DM Methodology is depicted in the diagram below, along with the measures involved.

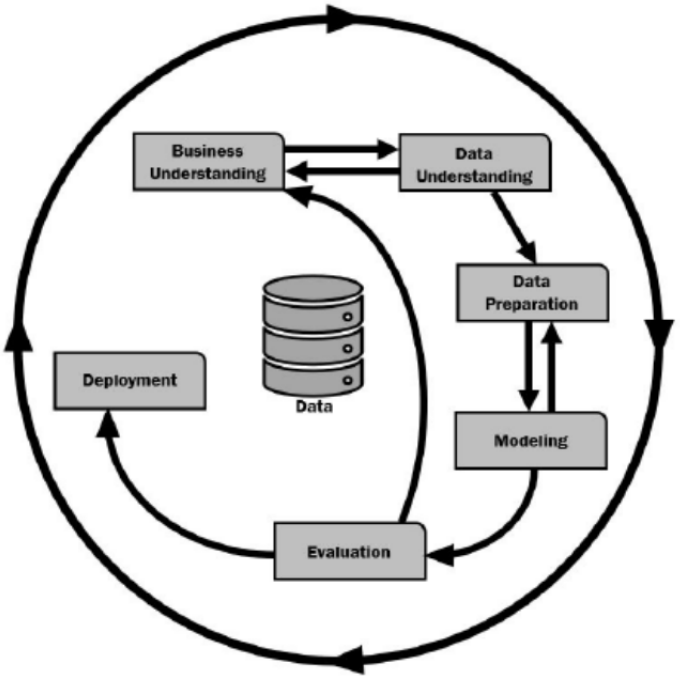


Figure 3: CRISP-DM Methodolog

2.4 Implementation

1. The Keras library is used to implement the final design of our RNN model in Python.

2. For our data, the model uses a fine-tuned Word Embedding Matrix, a ReLu activation function within the hidden layers, and a Sigmoid Function for output.

3. Adam (a Stochastic Gradient Descent extension) is the optimization (training) algorithm used.

4. The Loss Function is the standard Binary Cross Entropy, which simply ensures that our model maximizes the likelihood of our training results.

5. Dropout is used to help prevent our model from being overfitted.

6.Instead of a single train-test break, we conduct a 5-fold cross-validation evaluation of our model. In other words, we:

i. Arbitrarily shuffle the dataset.

ii. Divide the data into a total of k classes.

iii. For each one-of-a-kind group:

a. Use the category as a reserve or a test data set. b. As a training data collection, use the remaining classes. c. Fit a model to the training data and test it on the test data. d. Keep the test score but toss out the model.

iv. Finally, using the sample of model evaluation ratings, summarize the model's abilities.

2.5 Data Modeling: Recurrent Neural Network/LSTM

1.We created a bi-directional Recurrent Neural Network model with LSTM (Long Short Term Memory) modules and Attention Layers, based on the latest research in Natural Language Processing and Deep Learning.

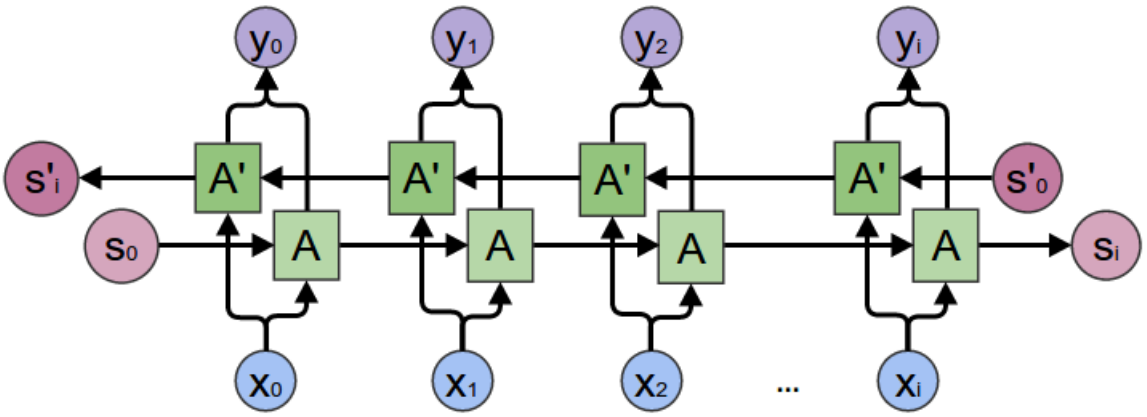


Figure 4: Bidirectional RNN

2.We wanted to try to replicate the acclaimed success in the Quora Challenge with this dynamic model, which has achieved state-of-the-art results in other NLP tasks.

3.The implemented model's bi-directional nature is suitable for this task because we want our RNN to learn representations from previous time steps as well as future time steps in the learning procedure.

4. Consider the following question: "Why is the Italian football team so bad?" vs. "Why is Rome the capital of Italy?" Since it is attempting to make a point, the first question will almost definitely be flagged as "Insincere" by Quora's guidelines. On the other hand, the second question is definitely correct. Given that each of these questions begin with the text "Why is the Italian...", a typical RNN, even with an LSTM implementation, would have difficulty accurately identifying them.

5.To put it another way, a standard RNN/LSTM will ignore future word sequences, which encapsulate crucial semantic data for the Quora Challenge. A bi-directional network, on the other hand, will relay back instances of words seen further down the line, resolving the issue.

3 Model Evaluation: RNN/LSTM Results

1.As can be shown, our model achieves an F1 Score of 0.67 across the four Epochs for which it was prepared, with a 96 percent accuracy.

2.We can safely say that the added difficulty, as compared to other modeling methods (such as logistic regression and the nave approach), was well worth it. Since the model achieves a significantly higher F1 score while only slightly increasing overall accuracy.

3.Given that Quora isn't interested in inferring or interpreting what constitutes a "sincere" or "insincere" query, and that the company determines this a priori, we can suggest our "black box" approach as an excellent tool for improving Quora's platform and overall business.

4 CONCLUSION

For this project our goal was to classify Quora questions as sincere or insincere. We experimented with a number of classification models and optimizations, with mostly positive results. Lstm with Adam optimization and updated decision boundary, as well as regularized Logistic Regression with an adjusted decision boundary, were some of the models we used.

4.1 Advantages of your approach

1.LSTM's ability to bridge very long time lags in the event of a problem is due to continuous error backpropagation within memory cells.

2.LSTM can deal with noise, distributed representations, and continuous values, among other things. Unlike finite state automata or hidden Markov models, LSTM does not require a finite number of states to be chosen a priori. It can deal with an infinite number of states in theory.

3.LSTM is effective across a wide range of parameters, including learning rate, input gate bias, and output gate bias. For example, the learning rates used in our experiments may seem high to some readers. A high learning rate, on the other hand, forces the output gates towards zero, automatically counteracting its own negative effects.

4.2 limitation

1.LSTMs gained popularity as a solution to the problem of vanishing gradients. However, it turns out that they were unable to fully delete it. The issue is that the data must also be moved from cell to cell in order to be evaluated. Furthermore, with the inclusion of additional features (such as forget gates), the cell has become very complex.

2.Getting educated and ready for real-world applications takes a lot of money and time. In technical terms, they require a large memory bandwidth due to the presence of linear layers in each cell, which the device typically lacks. As a result, LSTMs become unreliable in terms of hardware.

3.As data mining becomes more common, developers are looking for a model that can remember past data for a longer period of time than LSTMs. The human habit of splitting a given piece of knowledge into small parts for easy remembrance is the source of inspiration for such a model.

4.LSTMs are influenced by various random weight initializations and thus behave similarly to a feed-forward neural network. Instead, they choose small weight initializations.

5.LSTMs are susceptible to overfitting, and using the dropout algorithm to prevent this is difficult. When training a network, dropout is a regularization approach in which input and recurrent connections to LSTM units are probabilistically removed from activation and weight updates.

4.3 future work

There are a number of things we'd like to try if we were to pursue this project in the future. On the RNN side of things, we could extract a lot more features to try to enhance our results. Finally, with more time and computational resources (more efficient GPUs), we might see what happens with BERT with more data and training epochs, which would be interesting considering that BERT showed promising results even with limited data and time.

5 REFERENCES

1] C. Liu, Y. Sheng, Z. Wei, and Y. Yang, "Research of textclassi[U+FB01]cation based on improved tf-idf algorithm," in 2018IEEE International Conference of Intelligent Robotic andControl Engineering (IRCE), Aug 2018, pp. 218–222.

[2] O. Aborisade and M. Anwar, "Classi[U+FB01]cation for authorship oftweets by comparing logistic regression and naive bayes classi-[U+FB01]ers," in 2018 IEEE International Conference on InformationReuse and Integration (IRI), July 2018, pp. 269–276.

[3] S. T. Indra, L. Wikarsa, and R. Turang, "Using logistic regres-sion method to classify tweets into the selected topics," in 2016International Conference on Advanced Computer Science andInformation Systems (ICACSIS), Oct 2016, pp. 385–390.

[4] S.-J. Yen, Y.-S. Lee, J.-C. Ying, and Y.-C. Wu, "A logisticregression-based smoothing method for chinese text catego-rization." Expert Systems With Applications, vol. 38, pp. 11 581– 11 590, 2011.

[5] A. Onan, S. Koruko˘glu, and H. Bulut, "Ensemble of keywordextraction methods and classi[U+FB01]ers in text classi[U+FB01]cation." ExpertSystems With Applications, vol. 57, pp. 232 – 247, 2016.

[6] and, "A comparison of several ensemble methods for textcategorization," in IEEE International Conference onServicesComputing, 2004. (SCC 2004). Proceedings. 2004, Sep. 2004,pp. 419–422.