NLP Final Project – Quora Insincere Questions Classification

Yue Guo
yueguo@iu.edu
Indiana University
107 S. Indiana Avenue
Bloomington, IN 47405-7000

Jinju Jiang
heljiang@iu.edu
Indiana University
107 S. Indiana Avenue
Bloomington, IN 47405-7000

ABSTRACT

Quora is a platform that empowers people to learn from each other. On Quora, people can ask questions and connect with others who contribute unique insights and quality answers. A key challenge is to weed out insincere questions to keep their platform a place where users can feel safe sharing their knowledge with the world.In this project, we used text processing technologies to clean data, then create text classification model with LTSM, word embedding and so on. Our model obtained about 95/

KEYWORDS

Quora, Insincere Questions, LSTM, Word embedding, Nature Language Processing, text preprocessing, data cleaning

1 INTRODUCTION

"Quora is a question-and-answer website where questions are asked, answered, edited, and organized by its community of users in the form of opinions"[9]. Basically, on Quora, people can easily ask any questions and get some information they want. However, someone asks some insincere questions and someone intend to make a statement rather than wait for real answers. For example, someone asked "are Jat and Gujjar girls beautiful?" and another person asked "why Chinese people are always not welcome in all countries?". Those are all aggressive and offensive questions and we need to build a model to identify and flag insincere questions.

1.1 Motivation

Any question that assumes something debatable to be true, is insincere. For example, someone asked "why Chinese people are always not welcome in all countries?", they presupposed Chinese people are not welcome in all countries is always true. However, for questions on Quora and similar online public platform, they should be sincere attempts to obtain reusable information. Those people asking a question should be to obtain information, not to make a statement. Therefore, those questions that presuppose certain answers are really bad, not useful and even offend people. "Quora has employed both machine learning and manual review to address this problem"[10]. And we can develop a more scalable method to classify the insincere questions, and builder a better community for people sharing knowledge.

1.2 Background

The project is based on Quora, a question-and-answer website. People can ask, answer, edit, and organize questions there. And Quora wants to make sure that those questions are people interested in and could be able to answer those questions. Besides, both questions and answers should be in high quality, for example, those people answering the questions should be an expert in the topic. What's more, those people asking a question should be to obtain information, not to make a statement. Therefore, people really want to get answers from the Quora platform, but not presuppose certain answers. And there are some features that can signify that a question is insincere. First, it has a non-neutral tone to make a statement about a specific group of people; Second, it makes some insults against a specific group of people; Third, it is based on some false information; Last, it uses sexual content for shock value not to seek genuine answers.

And this kind of problem does not affect only online question-and-answer website, but also some other major website. If people see toxic and divisive content on those websites, they may not trust that website anymore. "Online ad giants Facebook and Google have increasingly found themselves on the hook for enabling the spread of socially divisive, offensive and at times out-and-out illegal content via their platforms fi?! in no small part as a consequence of the popularity of their content-sharing hubs"[12].

1.3 Related Work

In the academic world, there were lots of different kinds of research on Machine Learning skill in identifying insincere questions. Nikhil Garg is a Machine Learning engineer at Quora since 2012, and he introduced Machine Learning work at Quora. They have data based on interaction history, which is "highly engaged users(long history of activity e.g search queries, upvotes etc.) and ever-green content(long history of users engaging with the content in search, feed etc.)"[11]. He introduced four ways to identify good questions, such as duplicate question detection, answer ranking, topic expertise detection and moderation. First, they need to "detect duplicates even before question reaches the system"[11]. There are several reasons for them to delete duplicate questions, for example, people who are able to answer the question may be divided; no single question page becomes a best resource; those people looking for answers have to read many different question pages with similar question. Then, they used hand-labeled data, and build random forests, GBDT, deep neural networks models. Mihai Surdeanu in Yahoo! Research had a paper on rank answers on the Large Online question-answer collection, they "investigated a wide range of feature types, some exploiting NLP processors, and demonstrate that using them in combination leads to considerable improvements in accuracy"[14]. However, there are still some problems exist, for example, expert answers rank lower than popular writers' answers and those popular answers do not mean they are factually correct. And there are also some moderation challenges, such as spam, hate-speech, porn, plagiarism etc. Besides, some account use fake names, impersonation, sockpuppets.

1.4 Contributions

If we can make the question-answer website more sincere, more expert people will answer questions. Otherwise, if the website is full of spam, hate-speech, porn, plagiarism or some incorrect statement, then those really want to ask questions and share knowledge people will not use this website anymore. We can grow and share the Worldfis knowledge. If we build models that identify and flag insincere questions, we can help Quora uphold their policy of fiBe Nice, Be Respectfulfi and continue to be a place for sharing and growing the worldfis knowledgefi[10].

2 NLP APPROACH AND IMPLEMENTATION

2.1 Data

In this project,1.31 million sample data is available for training, and 56.4k rows of data is for testing.The training data includes the question that was asked, and whether it was identified as insincere (target = 1). the test data includes question id and question description. The relation for available train sample and test data is as the following 2 tables:

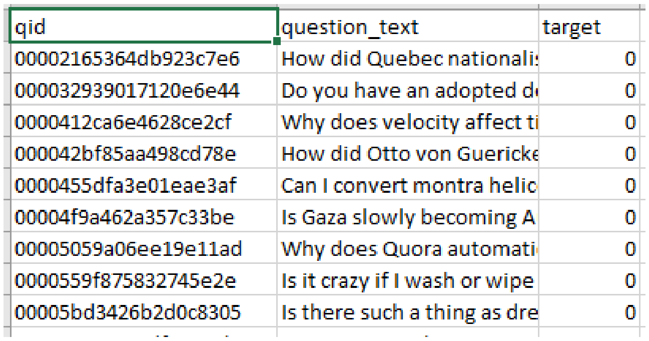
Table 1: Train table

Attributes	Description
qid	unique question identifier
question_text	Quora question text
target	a question labeled "insincere" has a value of 1, otherwise 0

Table 2: Test table

Attributes	Description
qid	unique question identifier
question_text	Quora question text

And here it is how training sample looks like:



qid	question_text	target
00002165364db923c7e6	How did Quebec nationali	0
000032939017120e6e44	Do you have an adopted d	0
0000412ca6e4628ce2cf	Why does velocity affect ti	0
000042bf85aa498cd78e	How did Otto von Guericke	0
0000455dfa3e01eae3af	Can I convert montra helic	0
00004f9a462a357c33be	Is Gaza slowly becoming A	0
00005059a06ee19e11ad	Why does Quora automati	0
0000559f875832745e2e	Is it crazy if I wash or wipe	0
00005bd3426b2d0c8305	Is there such a thing as dre	0

In addition to training data sample and test data, there were also some third party word embedding dataset:

- GoogleNews-vectors-negative300:
  https://code.google.com/archive/p/word2vec/
- glove.840B.300d:
  https://nlp.stanford.edu/projects/glove/
- paragram_300_sl999:
  https://cogcomp.org/page/resource_view/106
- wiki-news-300d-1M:
  https://fasttext.cc/docs/en/english-vectors.html

2.2 Approach and methodology

There are 3 main groups of approaches to solving NLP tasks.[8]

2.2.1 Rule-based. Rule-based approaches are the oldest approaches to NLP. Why are they still used, you might ask? It's because they are tried and true, and have been proven to work well. Rules applied to text can offer a lot of insight: think of what you can learn about arbitrary text by finding what words are nouns, or what verbs end in -ing, or whether a pattern recognizable as Python code can be identified. Regular expressions and context free grammars are textbook examples of rule-based approaches to NLP.

Rule-based approaches:

- tend to focus on pattern-matching or parsing
- can often be thought of as "fill in the blanks" methods
- are low precision, high recall, meaning they can have high performance in specific use cases, but often suffer performance degradation when generalized

2.2.2 "Traditional" Machine Learning. "Traditional" machine learning approaches include probabilistic modeling, likelihood maximization, and linear classifiers. Notably, these are not neural network models (see those below).

Traditional machine learning approaches are characterized by:

- training data - in this case, a corpus with markup
- feature engineering - word type, surrounding words, capitalized, plural, etc.
- training a model on parameters, followed by fitting on test data (typical of machine learning systems in general)
- inference (applying model to test data) characterized by finding most probable words, next word, best category, etc.
- "semantic slot filling"

2.2.3 Neural Networks. This is similar to "traditional" machine learning, but with a few differences:

- feature engineering is generally skipped, as networks will "learn" important features (this is generally one of the claimed big benefits of using neural networks for NLP)
- instead, streams of raw parameters ("words" – actually vector representations of words)

without engineered features, are fed into neural networks
- very large training corpus

Specific neural networks of use in NLP include recurrent neural networks (RNNs) and convolutional neural networks (CNNs).
In this project, we will combine "traditional" machine learning approach together with neural networks approach.

2.3 Implementation

2.3.1 Insincere question. In this project, Quora provides some standards to identify insincere question.[2] An insincere question is defined as a question intended to make a statement rather than look for helpful answers. Some characteristics that can signify that a question is insincere:

- Has a non-neutral tone
- Has an exaggerated tone to underscore a point about a group of people
- Is rhetorical and meant to imply a statement about a group of people
- Is disparaging or inflammatory
- Suggests a discriminatory idea against a protected class of people, or seeks confirmation of a stereotype
- Makes disparaging attacks/insults against a specific person or group of people
- Based on an outlandish premise about a group of people
- Disparages against a characteristic that is not fixable and not measurable
- Isn't grounded in reality
- Based on false information, or contains absurd assumptions
- Uses sexual content (incest, bestiality, pedophilia) for shock value, and not to seek genuine answers

2.3.2 Exploratory Data Analysis.

- Target distribution:6.2% of train questions are insincere.
- Top 20 1-gram words in sincere and insincere questions are as the following:Sincere questions are dominated by words like best, will, people, good, one, etc. with no reference to any specific nouns. Some of these words are high even in insincere words - meaning they are not significant to the classification.

  Insincere questions are dominated by words like trump, women, white, men, indian, muslims, black, americans, girls, indians, sex and india. More reference to specific groups of people of directly a person i.e. Donald Trump.

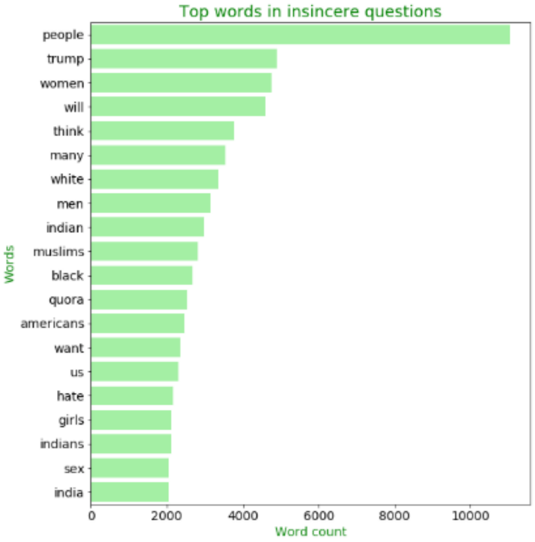


Figure 1: Top 20 1-gram words in insincere questions

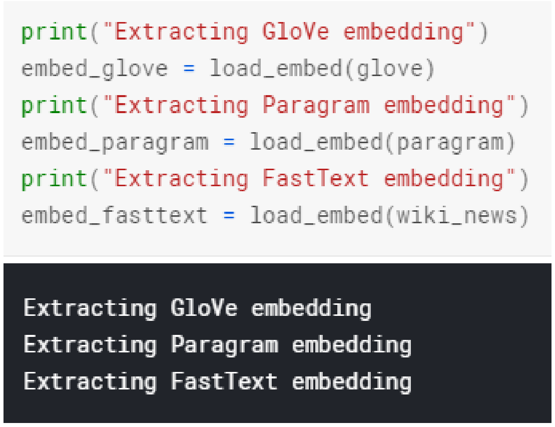
- Top 20 2-gram words in sincere and insincere questions:Top 3 bigrams in the insincere questions are 'Donald Trump', 'White People' and 'Black People'. Questions related to race are highly insincere.
  Presence of Chinese people, Indian muslims, Indian girls, North Indians, Indian women and White Women confirm the same.
  Sincere questions have best way, year old, will happen, etc. as the top ones. No clear trend there but 'best' is the key word to look for.
- Top 20 3-gram words in sincere and insincere questions:Insincere questions are related to hypothetical scenarios, age, race,

etc
Sincere questions are related to tips, advices, suggestions, facts, etc.

2.3.3 *Data Cleaning and Text prepossessing.*

- Data cleaning: When dealing with real-world data, dirty data is the norm rather than the exception. We continuously need to predict correct values, impute missing ones, and find links between various data artefacts such as schemas and records. We need to stop treating data cleaning as a piecemeal exercise (resolving different types of errors in isolation), and instead leverage all signals and resources (such as constraints, available statistics, and dictionaries) to accurately predict corrective actions. The primary goal of data cleaning is to detect and remove errors and anomalies to increase the value of data in analytics and decision making. While it has been the focus of many researchers for several years, individual problems have been addressed separately. These include missing value imputation, outliers detection, transformations, integrity constraints violations detection and repair, consistent query answering, deduplication, and many other related problems such as profiling and constraints mining.[1]
- Data Preprocessing is a technique that is used to convert the raw data into a clean data set. In other words, whenever the data is gathered from different sources it is collected in raw format which is not feasible for the analysis.I start with two golden rules:
  1. Don’t use standard preprocessing steps like stemming or stopword removal when you have pre-trained embeddings. Some of you might used standard preprocessing steps when doing word count based feature extraction (e.g. TFIDF) such as removing stopwords, stemming etc. The reason is simple: You loose valuable information, which would help you to figure things out.
  2. Get your vocabulary as close to the embeddings as possible I will focus in this notebook, how to achieve that. For an example I take the GoogleNews pretrained embeddings, there is no deeper reason for this choice.ords in insincere questions.

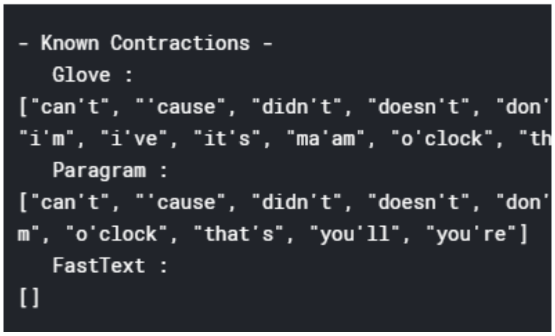
- using embedding data as provided in the project.



```
print("Extracting GloVe embedding")
embed_glove = load_embed(glove)
print("Extracting Paragram embedding")
embed_paragram = load_embed(paragram)
print("Extracting FastText embedding")
embed_fasttext = load_embed(wiki_news)
```

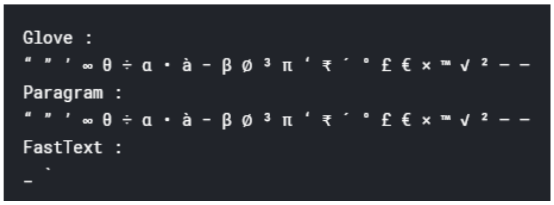
```
Extracting GloVe embedding
Extracting Paragram embedding
Extracting FastText embedding
```

- contractions transform: there are many words are written in contractions format such as ”aren’t” and ”can’t”. So we did contractions tranform by using predefined dictionary.



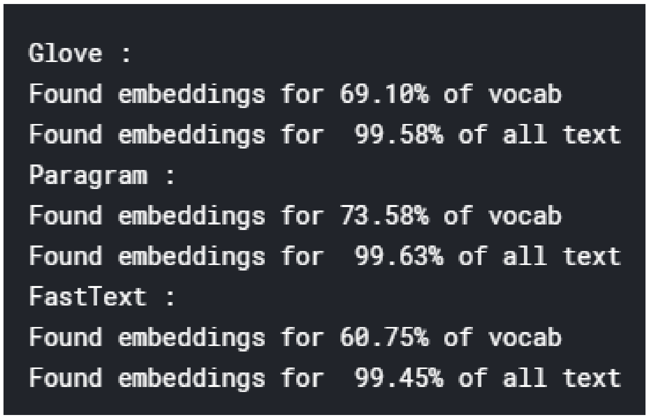
```
- Known Contractions -
   Glove :
["can't", "'cause", "didn't", "doesn't", "don'
"i'm", "i've", "it's", "ma'am", "o'clock", "th
   Paragram :
["can't", "'cause", "didn't", "doesn't", "don'
m", "o'clock", "that's", "you'll", "you're"]
   FastText :
[]
```

- special characters: there are a lot special characters which prevent recognition. We use a map to replace unknown characters with known ones and make sure there are spaces between words and punctuation.



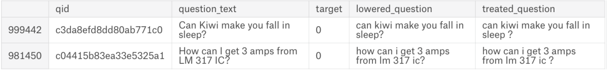
```
Glove :
“ ” ’ ∞ θ ÷ α • à − β ø ³ π ‘ ₹ ´ ° £ € × ™ √ ² — –
Paragram :
“ ” ’ ∞ θ ÷ α • à − β ø ³ π ‘ ₹ ´ ° £ € × ™ √ ² — –
FastText :
_ `
```

Finally, we made our vocabulary as close to the embeddings as we can:



```
Glove :
Found embeddings for 69.10% of vocab
Found embeddings for  99.58% of all text
Paragram :
Found embeddings for 73.58% of vocab
Found embeddings for  99.63% of all text
FastText :
Found embeddings for 60.75% of vocab
Found embeddings for  99.45% of all text
```

After data cleanning and text preprocessing, we got new field which is called as ”treated questions”



	qid	question_text	target	lowered_question	treated_question
999442	c3da8efd8dd80ab771c0	Can Kiwi make you fall in sleep?	0	can kiwi make you fall in sleep?	can kiwi make you fall in sleep ?
981450	c04415b83ea33e5325a1	How can I get 3 amps from LM 317 IC?	0	how can i get 3 amps from lm 317 ic?	how can i get 3 amps from lm 317 ic ?

Figure 2: Some samples of the preprocessed data, with qid, question_text, target, lowered_question and treated_question.

Which contains, qid(unique question identifier), question_text(Quora question text), target(a question labeled ”insincere” has a value of 1, otherwise 0), lowered_question(all characters in question text turned to lower case) and treated_question(all words in question text turned to original word).

2.3.4 Word Embedding. “Word embedding is one of the most popular representations of document vocabulary”[4]. We talked about this in class, try to use a vector to represent cat and dog and we also talked about how to generate those vectors.

“Word2Vec is one of the most popular technique to learn word embeddings using a shallow neural network. It was developed by Tomas Mikolov in 2013 at Google”[4]. From Wikipedia’s definition, Word2Vec models are shallow, two-layer neural networks that are trained to reconstruct linguistic contexts of words. Word2Vec takes a large corpus of text as its input and produces hundreds of dimensions usually three hundred dimensions vector space, “with each unique word in the corpus being assigned a corresponding vector in the space”[13]. Word vectors are assigned in the vector space, where similar words that share common contexts in the corpus are located closely.

2.3.5 Neural Network. The original goal of the Neural Network was to solve problems in the same way as human beings. We also talked Neural Network idea in class, although we did not analyze the backpropagation. Over time, neural networks have been used on a variety of tasks, including computer vision, speech recognition, social network filtering, and medical diagnosis.

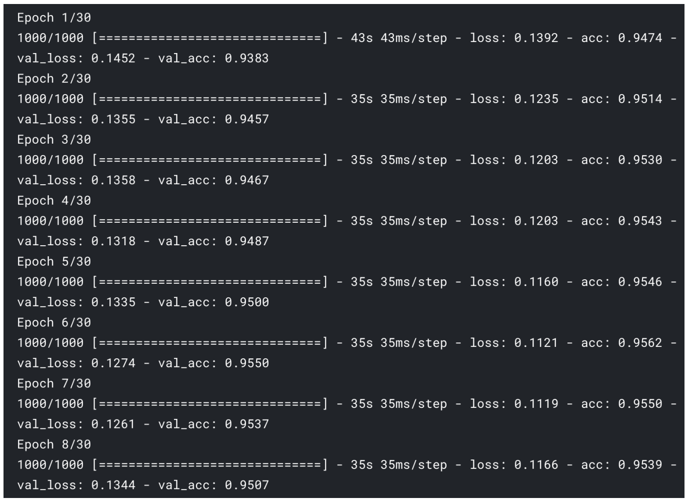
“The connections between artificial neurons are called ’edges’. Artificial neurons and edges typically have a weight that adjusts as learning proceeds”[3]. We increase or decreases the weight to change the strength of the signal at a connection. Typically, artificial neurons are aggregated into layers. Different layers can have different kinds of transformations on their inputs. Signals may go from the first layer (the input layer), then traverse the layers multiple times and finally to the last layer (the output layer).

Third, we need a model, here we consider a Neural Network and Recurrent Neural Network. “Long short-term memory (LSTM) units are units of a recurrent neural network (RNN). A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell”[5].

And to choose the model, I did lots of research online and found that there is an article about classifying Yelp review comments using deep learning techniques and word embeddings. And his situation is similar to ours, he used Yelp datasets to find out the positive or negative reviews. And for simplicity, he classified all reviews in two class, either as positive or negative. Therefore, the problem is a supervised learning just like us. And he used Keras library to build a neural network classifier. And also we found a kernel on Kaggle about this, and he also used LSTM as the main model. And he said that LSTM is all you need[6]. Therefore, I decided to learned LSTM more and use LSTM as our main model.

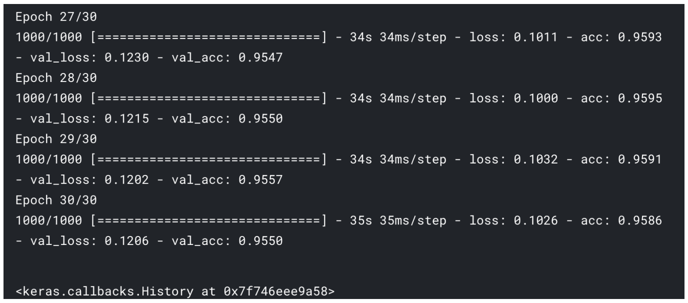
LSTM, which is short for Long Short-Term Memory, is a type of recurrent neural networks in deep learning filed. “In a simple way, LSTM networks have some internal contextual state cells that act as long-term or short-term memory cells. The output of the LSTM network is modulated by the state of these cells”[7]. The most important thing is it depends on the historical context of inputs, rather than only on the very last input. For example, if we want to write a book, it has Ross, Rachel, Monica and saw. We can say Ross saw Rachel, Rachel saw Monica, Monica saw Ross, and so on. We can easily make the model know that Ross, Rachel and Monica should be followed by saw. And saw and a dot could be after Ross, Rachel and Monica. However, we can still get “Ross saw Ross”, or “Ross.”, which are all bad. In LSTM, we can remember the words we used before, therefore, after we got Ross saw, we can only get Rachel or Monica as our output. LSTM is very good for translating text, turn speech to text and anything embedded in time.

As I followed those instructions online and built our own model. I built a three-layer model, the first two layers all have 64 units, and the last one uses the sigmoid as the activation. Following shows the accuracy after I trained 30 epochs.



```
Epoch 1/30
1000/1000 [==============================] - 43s 43ms/step - loss: 0.1392 - acc: 0.9474 -
val_loss: 0.1452 - val_acc: 0.9383
Epoch 2/30
1000/1000 [==============================] - 35s 35ms/step - loss: 0.1235 - acc: 0.9514 -
val_loss: 0.1355 - val_acc: 0.9457
Epoch 3/30
1000/1000 [==============================] - 35s 35ms/step - loss: 0.1203 - acc: 0.9530 -
val_loss: 0.1358 - val_acc: 0.9467
Epoch 4/30
1000/1000 [==============================] - 35s 35ms/step - loss: 0.1203 - acc: 0.9543 -
val_loss: 0.1318 - val_acc: 0.9487
Epoch 5/30
1000/1000 [==============================] - 35s 35ms/step - loss: 0.1160 - acc: 0.9546 -
val_loss: 0.1335 - val_acc: 0.9500
Epoch 6/30
1000/1000 [==============================] - 35s 35ms/step - loss: 0.1121 - acc: 0.9562 -
val_loss: 0.1274 - val_acc: 0.9550
Epoch 7/30
1000/1000 [==============================] - 35s 35ms/step - loss: 0.1119 - acc: 0.9550 -
val_loss: 0.1261 - val_acc: 0.9537
Epoch 8/30
1000/1000 [==============================] - 35s 35ms/step - loss: 0.1166 - acc: 0.9539 -
val_loss: 0.1344 - val_acc: 0.9507
```

Figure 3: The first part of my training result.



```
Epoch 27/30
1000/1000 [==============================] - 34s 34ms/step - loss: 0.1011 - acc: 0.9593
- val_loss: 0.1230 - val_acc: 0.9547
Epoch 28/30
1000/1000 [==============================] - 34s 34ms/step - loss: 0.1000 - acc: 0.9595
- val_loss: 0.1215 - val_acc: 0.9550
Epoch 29/30
1000/1000 [==============================] - 34s 34ms/step - loss: 0.1032 - acc: 0.9591
- val_loss: 0.1202 - val_acc: 0.9557
Epoch 30/30
1000/1000 [==============================] - 35s 35ms/step - loss: 0.1026 - acc: 0.9586
- val_loss: 0.1206 - val_acc: 0.9550


<keras.callbacks.History at 0x7f746eee9a58>
```

Figure 4: The final part of my training result.

2.4 Evaluation

Since the data is from kaggle, we cannot know the exact answer for the test cases. But since we have the rule from kaggle, we can identify whether a question is sincere or not. Here is the rule from kaggle[10],

Has a non-neutral tone

Has an exaggerated tone to underscore a point about a group of people

Is rhetorical and meant to imply a statement about a group of people

Is disparaging or inflammatory

Suggests a discriminatory idea against a protected class of people, or seeks confirmation of a stereotype

Makes disparaging attacks/insults against a specific person or group of people

Based on an outlandish premise about a group of people

Disparages against a character that is not fixable and not measurable

Isn't grounded in reality

Based on false information, or contains absurd assumptions

Uses sexual content (incest, bestiality, pedophilia) for shock value, and not to seek genuine answers

2.5 Timeline

First, Hellen did research online how to preprocess data and did research online to choose which word embeddings library should choose. Then she figure out how to use that library to turn our question sentences to vectors and use them as input. Furthermore, Yue tried different Machine Learning models, such as LSTM, Neural Network, and find which algorithms to train the model is better. Finally, we apply the test data to my model.

Nov. 21 - Dec. 1: Get data from kaggle; Choose a embeddings library, and turn questions to vectors;

Dec. 1 - Dec. 7: Preprocess data; Train model on GPU;

Dec. 7 - Dec. 10: Apply test data on model;

Dec. 10 - Dec. 13: Get final report done.

2.6 Further work

We found that LSTM really works very well on text and something embedded in time. And for machine learning models, not only the algorithms and models we use is important, but also data preprocessing. And if we have more time, we can change all different kind of word embedding resources and check whether the result and the accuracy has any different. Besides, we can try more different deep learning models.

For the future, we should take some other factors in consideration. And we also should help to identify the sincere answers. Since lots of people also answer very bad. We need to know the answers really answer the question, is factually correct, is clear and easy to read, supported with rationale and demonstrates credibility.

Furthermore, we can try to find out whether people those who answer the same questions show enough respect to each other, or are they having a big fight online.

REFERENCES

[1] [n. d.]. data cleaning = Web, year=2018, note = Accessed: 2018-12-10 https://en.wikipedia.org/wiki/Data_cleansing,. ([n. d.]).

[2] [n. d.]. General Description for insincere question = Web, year=2018, note = Accessed: 2018-12-10 https://www.kaggle.com/c/quora-insincere-questions-classification/data,. ([n. d.]).

[3] 2018. Artificial neural network. Web. (2018). Accessed: 2018-12-08 https://en.wikipedia.org/wiki/Artificial_neural_network.

[4] 2018. Introduction to Word Embedding and Word2Vec. Web. (2018). Accessed: 2018-12-08 https://towardsdatascience.com/introduction-to-word-embedding-and-word2vec-652d0c2060fa.

[5] 2018. Long short-term memory. Web. (2018). Accessed: 2018-12-08 https://en.wikipedia.org/wiki/Long_short-term_memory.

[6] 2018. LSTM is all you need! Web. (2018). Accessed: 2018-12-12 https://www.kaggle.com/mihaskalic/lstm-is-all-you-need-well-maybe-embeddings-also.

[7] 2018. The magic of LSTM neural networks. Web. (2018). Accessed: 2018-12-12 https://medium.com/datathings/the-magic-of-lstm-neural-networks-6775e8b540cd.

[8] 2018. The Main Approaches to Natural Language Processing Tasks. Web. (2018). Accessed: 2018-12-10 https://www.kdnuggets.com/2018/10/main-approaches-natural-language-processing-tasks.html.

[9] 2018. Quora. Web. (2018). Accessed: 2018-12-05 https://en.wikipedia.org/wiki/Quora.

[10] 2018. Quora Insincere Questions Classification Description. Web. (2018). Accessed: 2018-12-05 https://www.kaggle.com/c/quora-insincere-questions-classification.

[11] 2018. Scaling Quality On Quora Using Machine Learning. Web. (2018). Accessed: 2018-12-06 https://qconsf.com/sf2016/system/files/presentation-slides/scaling_quality_using_machine_learning_-_qcon_sf_2016.pdf.

[12] 2018. Unilever warns social media to clean up fitoxicfi content. Web. (2018). Accessed: 2018-12-12 https://techcrunch.com/2018/02/12/unilever-warns-social-media-to-clean-up-toxic-content/.

[13] 2018. Word2Vec. Web. (2018). Accessed: 2018-12-08 https://en.wikipedia.org/wiki/Word2vec.

[14] Mihai Surdeanu, Massimiliano Ciaramita, and Hugo Zaragoza. 2008. Learning to Rank Answers on Large Online QA Collections. In *ACL*.