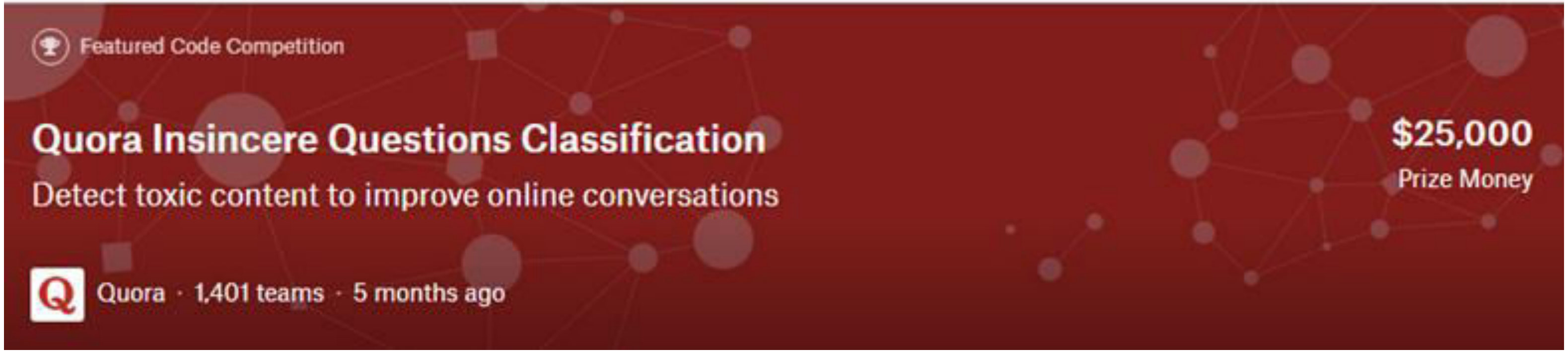
Quora Insincere Questions Classification

Nguyễn Thành Long & Nguyễn Bá Nghĩa

Introduction & Overview of the problem



Featured Code Competition

Quora Insincere Questions Classification

Detect toxic content to improve online conversations

Quora · 1,401 teams · 5 months ago

$25,000

Prize Money

An existential problem for any major website today is how to handle toxic and divisive content. Quora is a platform to gain & share knowledge where you can ask any question and get answers from different people with unique insights. At the same time, it’s important to handle the toxic contents to make the users safe to share their knowledge.
kaggle.com/c/quora-insincere-questions-classification/data

Dataset the Task:

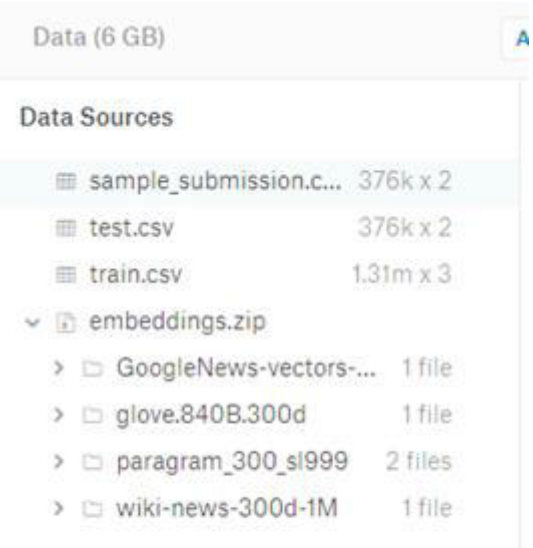
The task: Build the model to predict whether a question asked is sincere or insincere

Overview about dataset:

+Link to download dataset: https://www.kaggle.com/c/quora-insincere-questions-classification/data

+About this dataset:

- Embeddings.zip: Some pretrain embedding set (Glove, Word2Vec, FastText, Paragram)
- Train.csv: Trainning set with 1306122 samples is question that labeled to 0 (sincere) or 1 (insincere).
- Test.csv: Testing set includes 56370 samples is question without label



Data (6 GB)

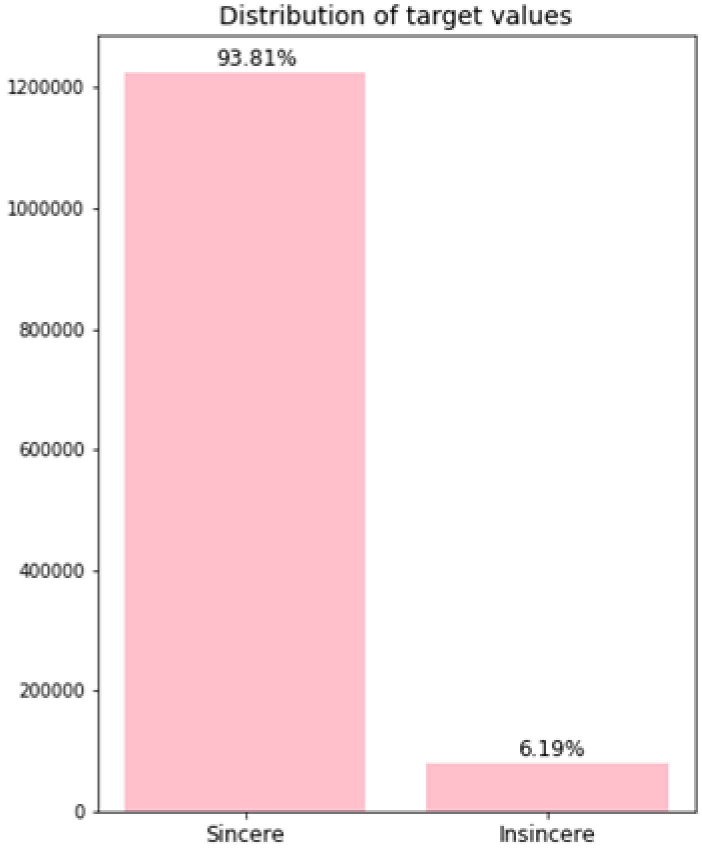
Data Sources

sample_submission.c...	376k x 2
test.csv	376k x 2
train.csv	1.31m x 3
embeddings.zip	
GoogleNews-vectors-...	1 file
glove.840B.300d	1 file
paragram_300_sl999	2 files
wiki-news-300d-1M	1 file

Exploratory Data Analysis:

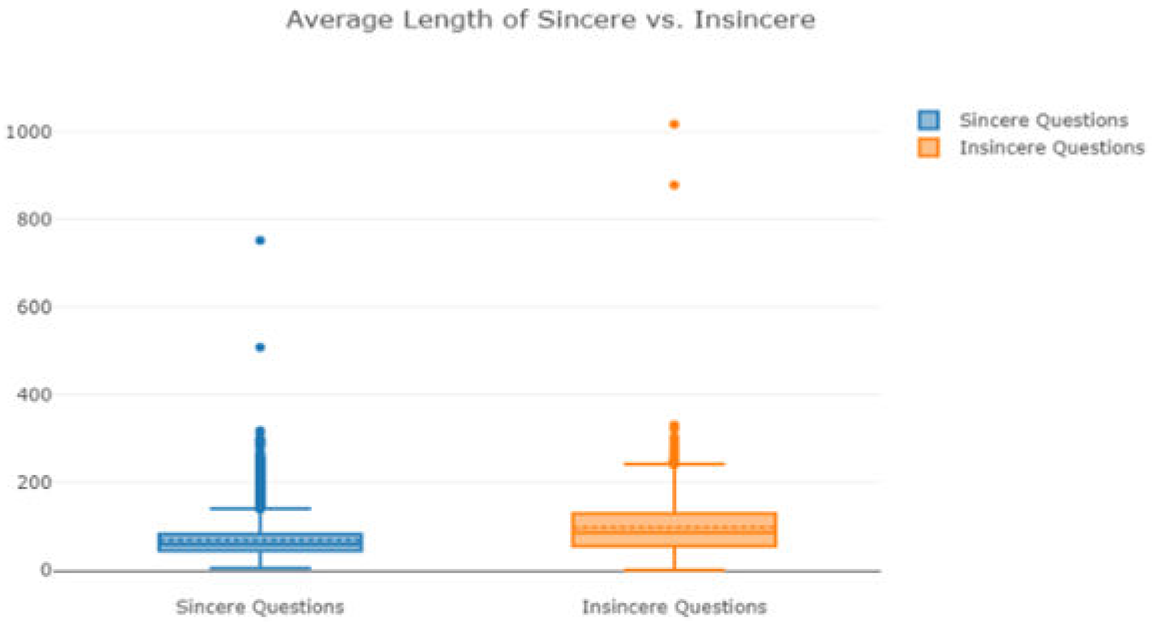
The distribution of target:

We can see that this data set is highly imbalance with only 6.19 percent of insincere question and 93.81 percent of sincere question. Resampling and data augmentation maybe improve model performance. Moreover, evaluation metric F1-score will be work in this case because it considers both precision and recall of the test to compute the score.



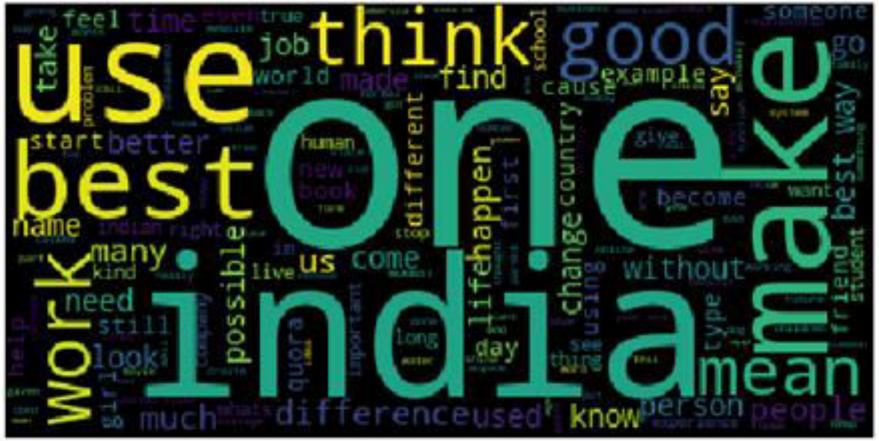
The length of question:

Seems like length doesn’t explain insincerity but we can see that the length of insincere questions are greater than sincere questions. Let’s check the maximum length question:

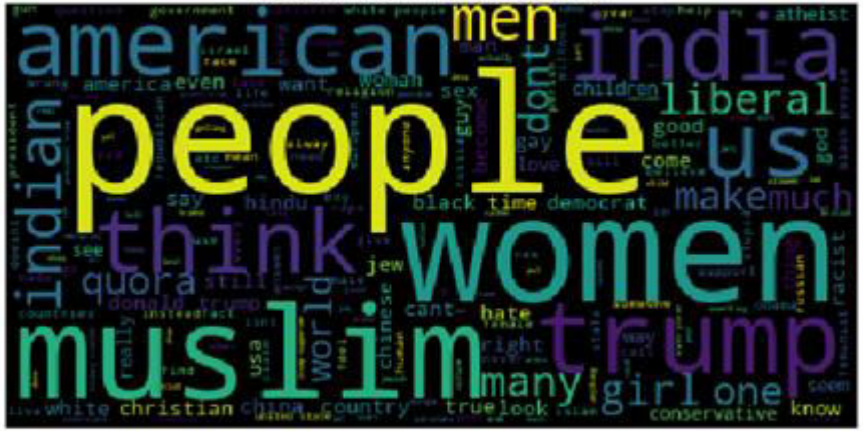


Check most used words in each class of questions:

Top used word in sincere question



Top used word in insincere question



Observations:

- Some of the top used words are common across both the classes likes 'think', 'india', ...
- The other top used words in sincere question after excluding the common ones are 'one', 'make', 'good', 'best', ...

- The other top used words in sincere question after excluding the common ones are 'one', 'make', 'good', 'best', ...

Model Selection

Try machine learning with Logistic Regression.

Feature extraction: Using TF-IDF.

Model: Logistic Regression CV.

Parameters:

- interception = True
- penalty = l2 norm
- solver = lbfgs
- max iteration = 20
- random state = 1

Result after training on test set:

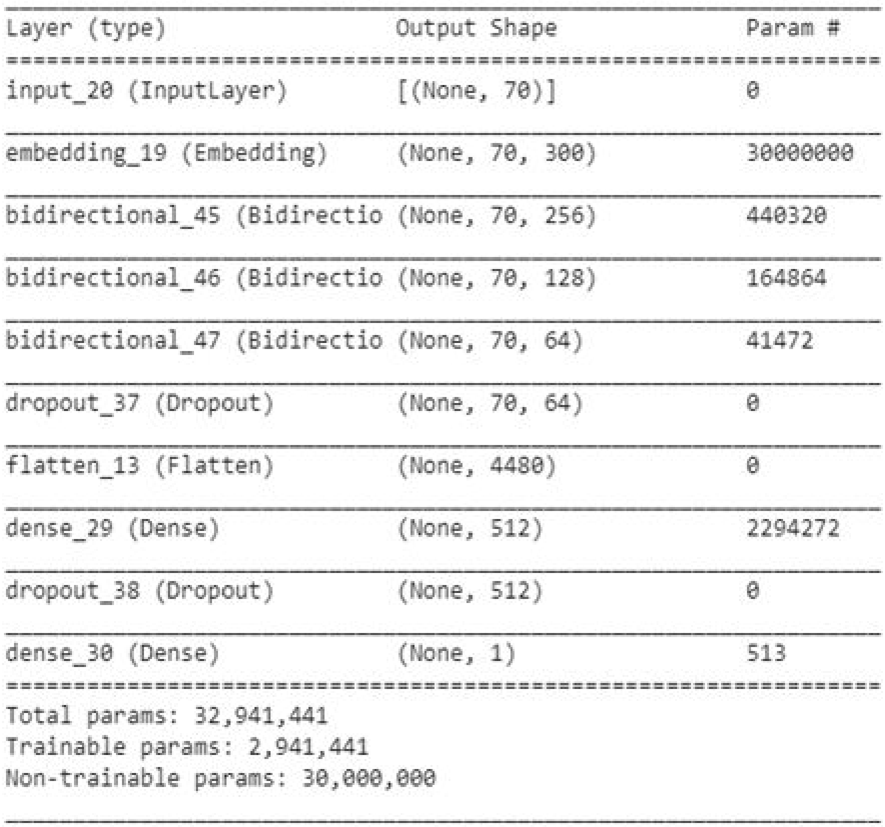
F1 score = 0.5485900545785324

Accuracy score = 0.9544071202985932

Using deep learning with BiLSTM.

Embedding: Using glove 300d set.
Padding to the end of question with max length = 70.

-First layer is embedding layer using weights of glove set and output shape is (n_samples, max_length, embedding dim)
-Second, 3th and 4th layer is three Bidirectional LSTM layers with sequence output dimentional is: 256, 128, 64
-After three BiLSTM layer, we using one Drop out layer to decrease connected of previor layer with probability = 0.2.
-Then we Flatten output of Drop out layer to shape (n_sample, dimensional)
-Next, we using 1 fully connected layer with relu as activation and output dimensional: 512
-We using Dropout layer again with prob = 0.2.
-Last, we using one Fully connected layer with sigmoid as activation function with output dimensional: 1.
-*Loss function:* Cross entropy loss



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_20 (InputLayer)        [(None, 70)]              0
_________________________________________________________________
embedding_19 (Embedding)     (None, 70, 300)           30000000
_________________________________________________________________
bidirectional_45 (Bidirectio (None, 70, 256)           440320
_________________________________________________________________
bidirectional_46 (Bidirectio (None, 70, 128)           164864
_________________________________________________________________
bidirectional_47 (Bidirectio (None, 70, 64)            41472
_________________________________________________________________
dropout_37 (Dropout)         (None, 70, 64)            0
_________________________________________________________________
flatten_13 (Flatten)         (None, 4480)              0
_________________________________________________________________
dense_29 (Dense)             (None, 512)               2294272
_________________________________________________________________
dropout_38 (Dropout)         (None, 512)               0
_________________________________________________________________
dense_30 (Dense)             (None, 1)                 513
=================================================================
Total params: 32,941,441
Trainable params: 2,941,441
Non-trainable params: 30,000,000
_________________________________________________________________
```

Training model:

Loss Function: Cross Entropy Loss

Optimization: Adam

Training Metric: Accuracy

Validation set = 0.2, Training set = 0.8

Number of epochs: 7

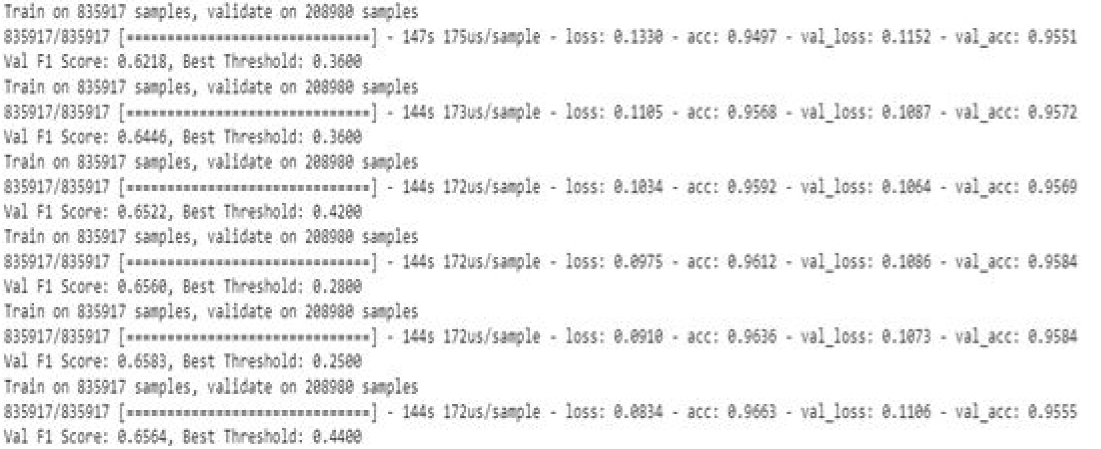
Batch size = 512.

Evaluation model:

Evaluation metric: F1 score.

We find the best threshold that give the max f1 score in range (0.1, 0.501) with step 0.01.

Result on training set:



```
Train on 835917 samples, validate on 208980 samples
835917/835917 [==============================] - 147s 175us/sample - loss: 0.1330 - acc: 0.9497 - val_loss: 0.1152 - val_acc: 0.9551
Val F1 Score: 0.6218, Best Threshold: 0.3600
Train on 835917 samples, validate on 208980 samples
835917/835917 [==============================] - 144s 173us/sample - loss: 0.1105 - acc: 0.9568 - val_loss: 0.1087 - val_acc: 0.9572
Val F1 Score: 0.6446, Best Threshold: 0.3600
Train on 835917 samples, validate on 208980 samples
835917/835917 [==============================] - 144s 172us/sample - loss: 0.1034 - acc: 0.9592 - val_loss: 0.1064 - val_acc: 0.9569
Val F1 Score: 0.6522, Best Threshold: 0.4200
Train on 835917 samples, validate on 208980 samples
835917/835917 [==============================] - 144s 172us/sample - loss: 0.0975 - acc: 0.9612 - val_loss: 0.1086 - val_acc: 0.9584
Val F1 Score: 0.6560, Best Threshold: 0.2800
Train on 835917 samples, validate on 208980 samples
835917/835917 [==============================] - 144s 172us/sample - loss: 0.0910 - acc: 0.9636 - val_loss: 0.1073 - val_acc: 0.9584
Val F1 Score: 0.6583, Best Threshold: 0.2500
Train on 835917 samples, validate on 208980 samples
835917/835917 [==============================] - 144s 172us/sample - loss: 0.0834 - acc: 0.9663 - val_loss: 0.1106 - val_acc: 0.9555
Val F1 Score: 0.6564, Best Threshold: 0.4400
```

Result on training set:

There is significant overfiting after 6 epochs when loss on validation set at 6th epoch greater than loss at 5th epoch while loss on training set continue decrease and model early stop after 6 epochs.

Result on Testing set:

- F1-Score: 0.6623 with threshold 0.49
- Accuracy score: 0.9569 with threshold 0.5