### **Introduction & Overview of the problem:**

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.

**Dataset the Task:**

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

*Overview about dataset:*

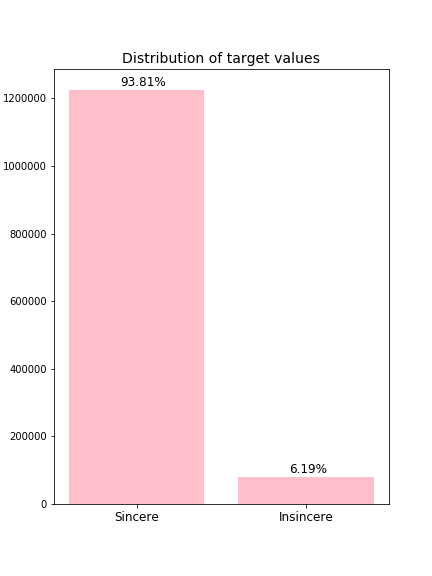
Link to download dataset: <https://www.kaggle.com/c/quora-insincere-questions-classification/data>

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

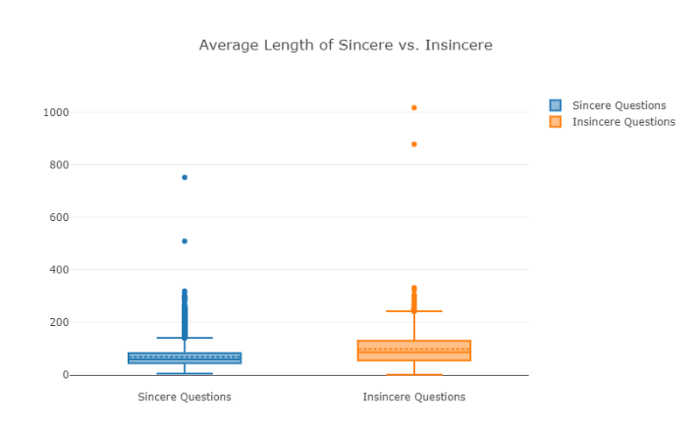
**Exploratory Data Analysis:**

**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:**

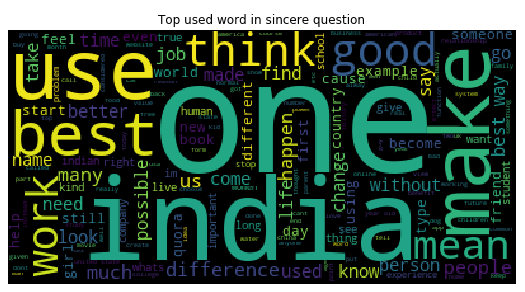
****

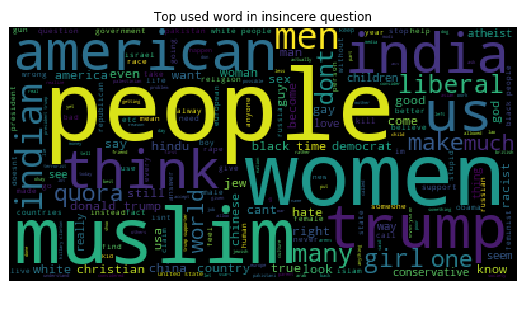
Seems like length doesn’t explain insincerity but we can see that the length of insincere questions are greater than sincere questions. Let’s check the maximum length question:

"What is [math]\\frac{\\int\_{1x^5}^{3x^{-5}} \\tan(\\tan({\\boxed{\\int\_{1x^0}^{1x^2} \\sum\_{\\varpi=1}^{\\infty} \\int\_{2x^{-3}}^{2x^2} \\sum\_{\\alpha=7}^{\\infty} \\underbrace{\\sqrt[2]{1x^5}}\_{\\text{Gauss's Law of Theoretical Probability.}} d\\tau dx}}^{1x^0})) d\\mu}{\\int\_{2x^{-3}}^{1x^5} \\cos(\\int\_{2x^2}^{1x^{-3}} \\frac{\\sqrt[2]{\\overbrace{\\underbrace{\\frac{3x^3+3x^5}{\\sqrt[3]{2x^{-3}}}}\_{\\text{Gauss's Law of Theoretical Probability.}} \\times \\overbrace{\\tan(2x^0)}^{\\text{Gauss's Law of Theoretical Probability.}}-\\sum\_{4=7}^{\\infty} \\boxed{3x^{-5}}}^{\\text{Inverse Function.}}}}{{\\boxed{\\int\_{2x^2}^{2x^4} 3x^1 d9} \\div \\sum\_{6=6}^{\\infty} \\sqrt[3]{2x^2}+\\sqrt[4]{\\sin(2x^0+3x^0)}}^{2x^{-4}}+\\boxed{\\frac{\\vec{\\boxed{\\sum\_{\\gamma=10}^{\\infty} 1x^{-5}}}}{\\frac{\\sum\_{\\iota=2}^{\\infty} 1x^{-5}-\\frac{3x^{-1}}{1x^{-4}}}{\\sin(\\tan(3x^{-2}))}}}} \\times \\boxed{\\sqrt[2]{{{{\\sqrt[5]{2x^5}}^{2x^{-1}}}^{2x^{-1}} \\div \\sum\_{\\chi=6}^{\\infty} \\int\_{1x^4}^{2x^{-4}} 3x^2 d\\vartheta+{2x^{-3}}^{2x^{-5}}}^{3x^{-4}}}} d\\mu) d\\iota}[/math]?"

* There is significant noise in our data.

**Check most used words in each class of questions:**

****

****

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’, ...

**Text Pre-processing:**

* Punctuation removal
* Stopword removal
* Tokenization

**Model Selection**

**1, 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

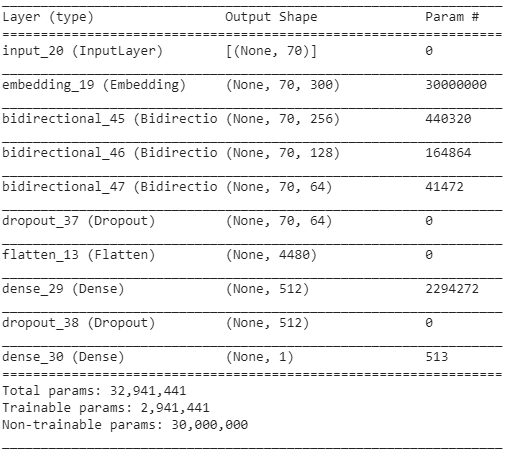
Accuracy score = 0.9544071202985932

**2, Using deep learning with BiLSTM.**

*Embedding:* Using glove 300d set.

*Padding to the end of question with max length = 70.*

*Model Sumary:*



* 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.

**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:** There is significant overfiting after 7 epochs when loss on validation set at 6th epoch is