Quora Insincere Questions Classification

Sanya Zaveri
AU1920064
sanya.z@ahduni.edu.in
Ahmedabad University

Mihir Pathak
AU1920138
mihir.p@ahduni.edu.in
Ahmedabad University

Malav Doshi
AU1940017
malav.d@ahduni.edu.in
Ahmedabad University

Parth Shah
AU1940065
parth.s5@ahduni.edu.in
Ahmedabad University

***Abstract*—The internet today has become an unrivalled source of information where people converse on content based websites such as Quora asking doubts and sharing knowledge with the world. A major arising problem with such websites is the proliferation of toxic comments or instances of insincerity wherein the users instead of maintaining a sincere motive indulge in spreading toxic and divisive content. The obvious choice in confronting this situation is detecting such content beforehand and preventing it from publishing online.**

***Keywords*—Text Classification, Tokenization, Logistic Regression, Lemmatization, Confusion Matrix**

I. Introduction

Quora is a website where a community of users can ask and answer questions. They may also create Q&A blogs and analyze statistics about the users as well. The relevance of question is often taken as into consideration and is necessary to prevent an unnecessary or redundant question that might affect different set of people.
Quora Insincere questions arises when people ask questions that is intended to make statement rather than look for helpful answers. Questions consisting of an non-neutral term, disparaging, inflammatory or is not grounded in reality can be classified as Insincere questions.

II. Related Work

There have been many contributions in the domain of text classification. It is due to the business value it adds by classifying text on the basis of various parameters. Another such paper proposed a model for classifying tweets [cite]. The author had implemented a Logistic Regression model of machine learning for classifying tweets according to their the topic they belong. The system transformed the tweets into vector which is acceptable by the model. The confusion matrix showed an accuracy of around 92%. This system uses just one algorithm which does not give any evidence that this model has performed the best. The proposed system will implement 4 models which will enable to analyze which is performing better.

III. Understanding the Data

A. Data Analysis

The Data consists of *test.csv, train.csv* and *embeddings.zip*. The train.csv consists of the question id, questions text and classification into *0* or *1*, where 0 denotes a sincere question and 1 denotes an insincere question. This csv file consists of about 1.3 million entries.
We use the following method for data analysis:

1) Bar-Graph
On visualising the given training set on a bar graph (Figure. 1), we observe the data has 1.3 million entries, in which 93% of entries are sincere questions, while the rest questions are insincere questions.

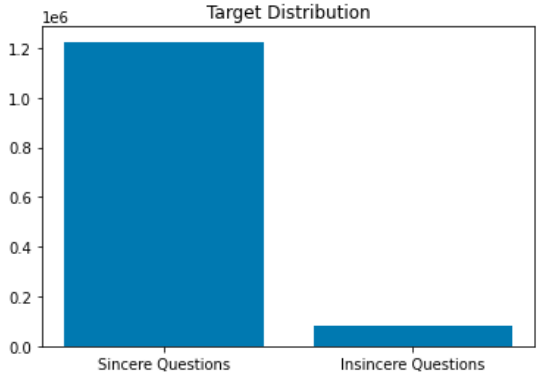

Fig. 1. Distribution of Questions into Sincere and Insincere

2) WordCloud
WordCloud helps us to observe nature of sincere and insincere questions. In Fig. 2. it is observed that ords such as *India and Indian*, which have essentially the same root, hence we need to apply lemmatization in our training set. Similarly words such as *will* shows the need for removing the stop words to only consider relevant words.

Fig. 2. WordCloud of Insincere Questions

B. Data Preprocessing

1) Removing Stop Words
   Stop words act like connectors in any sentence. These words do not add any value of their own. Examples include *a, the, is* and *are*. Removal of stop words helps us to remove low-level information and shifts our focus to value adding words. The NLTK library is used to remove stop words, includes the list of stop words for different languages.
2) Lemmatization
   Root words are the word from which other words are derived. For eg, *eat* is the root word for *eating, to eat* Reducing the inflected words while ensuring the root word belongs to the selected language (unlike Stemming) is known as Lemmatization. Wordnet Lemmatizer from the NLTK library is used for this purp

IV. Feature Engineering

We extract features from the raw data set using domain knowledge we have. We do this as having more defined features will help the model improve its' quality of results.
features added = ['num words','num unique words','len question text', 'len char question text', 'len word question text','num stopwords','common words','len mean words']

V. Modeling

For the problem statement we will be using a Logistic Regression model that is provided from the sklearn.linear model library. After feature engineering the data set is split into train and test in the ratio of 80% and 20% respectively. Logistic Regression function is used with the parameters, C = 0.1, solver = 'sag' and max iterations = 1000.
Where,

1) C = 0.1
   Inverse of regularization strength; must be a positive float. Like in support vector machines, smaller values specify stronger regularization.
2) solver = 'sag
   Stochastic Average Gradient descent. A variation of gradient descent and incremental aggregated gradient approaches that uses a random sample of previous gradient values. Fast for big data sets.
3) max iterations
   Maximum number of iterations taken for the solvers to converge.

After the model is fitted to the given data set, it is used to predict the labels for the unseen future data. X test and y test are compared to get accuracy of the model.

VI. Inferences

1) F1 Score
   a) Precision: It is ratio of correctly predicted outcomes to the total predicted outcomes. It is used in case of class imbalance.
   b) Recall: It is the ratio of correctly predicted outcomes to the total outcomes. It is also known as sensitivity of the model.

   It is calculated by computing the harmonic mean of precision and recall measures. When these values are closely related, F1 Score is the average of the two.
   We achieve a F1 Score of 0.91 on our model.
2) Confusion Matrix
   The original dataset has 1.3M data points. Here, we are taking a sample space of 13,000 data points for faster computation.
   a) Top Left - "true positive" for correctly predicted event values. (predicted 0, actual 0)
   b) Top Right - "false positive" for incorrectly predicted event values. (predicted 1, actual 0)
   c) False Negative - "true negative" for correctly predicted no-event values. (predicted 1, actual 1)
   d) True Negative - "false negative" for incorrectly predicted no-event values. (predicted 0, actual 1)

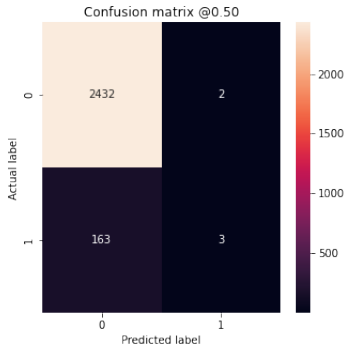

Fig. 3. Confusion Matrix on subspace

References

[1] S. T. Indra, L. Wikarsa and R. Turang, "Using logistic regression method to classify tweets into the selected topics," 2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS), 2016, pp. 385-390, doi: 10.1109/ICACSIS.2016.7872727.

[2] O. Aborisade and M. Anwar, "Classification for Authorship of Tweets by Comparing Logistic Regression and Naive Bayes Classifiers," 2018 IEEE International Conference on Information Reuse and Integration (IRI), 2018, pp. 269-276, doi: 10.1109/IRI.2018.00049.

[3] "Sklearn.linear_model.logisticregression," scikit. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html. [Accessed: 20-Mar-2022].

[4] "Precision-recall," scikit. [Online]. Available: https://scikit-learn.org/stable/auto_examples/model_selection/plot_precision_recall.html. [Accessed: 20-Mar-2022].

[5] Y. Liu, J. Niu, Q. Zhao, J. Lv and S. Ma, "A Novel Text Classification Method for Emergency Event Detection on Social Media," 2018 IEEE SmartWorld, Ubiquitous Intelligence Computing, Advanced Trusted Computing, Scalable Computing Communications, Cloud Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), 2018, pp. 1106-1111, doi: 10.1109/SmartWorld.2018.00192.

[6] "Quora insincere questions classification," Kaggle. [Online]. Available: https://www.kaggle.com/c/quora-insincere-questions-classification. [Accessed: 20-Mar-2022].