Quora Insincere Questions Classification

Abstract

Internet has opened up a new horizon of opportunities. It has answers to all questions, and even if answer is not found, there are sites where you can ask any question and people from all over the world answer it. Quora is one such question answer Here anyone can ask question from any domain, which are then answered by .website others. People misuse this boon and create a havoc by asking questions that are not to be asked in a public forum. In this project, we propose to develop a system that questions so that appropriate actions can be taken against ”recognizes these "insincere them. The questions are classified as insincere if they have a non-neutral tone, is disparaging, inflammatory. In this, project a model of text classification using Logistic Regression. Based on the models, Logistic Regression performed the best recall, , The metrics used for judging the result include F1 score,followed by TF-IDF accuracy and precision.

Design

This project originates from preprocessed train data from Quora Insincere Questions competition 2018.The data is provided by kaggle and presents to classify these insincere questions, and to remove stop words, numbers, punctuations, commons words and converted to lower case. So, that appropriate actions can be taken against them.

Data

The dataset is collected from a website known as Kaggle, it is a website which has many datasets that are available for public use. The dataset is available on kaggle, it contains approximately 1.3 million rows and 4 columns: index, questions id, question text and question type (insincere or sincere).

Algorithms

We start with cleaning the Dataset, first we Checked: missing, Null values, data formats duplicates and clean question text (remove numbers, punctuations, stop words and convert to lowercase), we changed the data type of the target to object, then drop un-necessary columns, remove white spaces and reindex the columns.

For "EDA" we interested to show the imbalance on target we use catplot to represent that and for identify frequent words in question text we use cloudword .

Molding

Logistic regression, Naive Bayes, TF-IDF. Logistic regression feature importance ranking was used directly to guide the choice and order of variables.

Model Evaluation and Selection

The entire training dataset contains approximately 1.3 records was split into 30 train.

The official metric for Dataset was classification rate (F1), then Using TF-IDF have improve the recall, but the accuracy and precision of the first logistic regression model still outperforms the other models.

Logistic regression score

- Accuracy 0.942
- F1 0.970
- precision 0.948
- recall 0.993

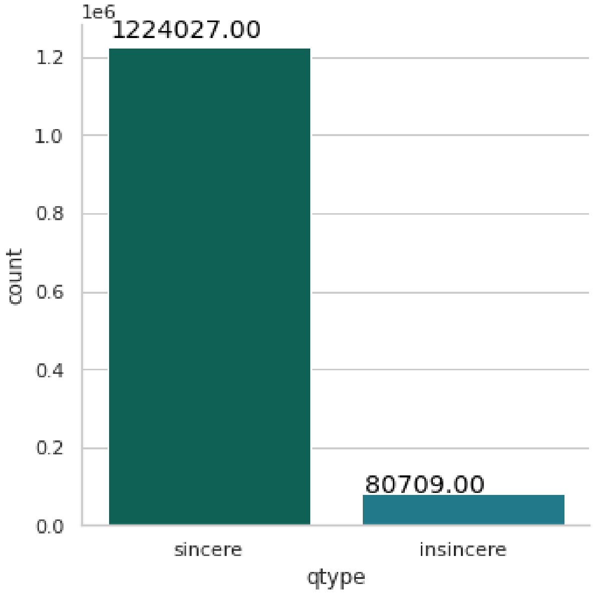
LR-TFIDF score

- Accuracy 0.941
- F1 0.969
- precision 0.947
- recall 0.993

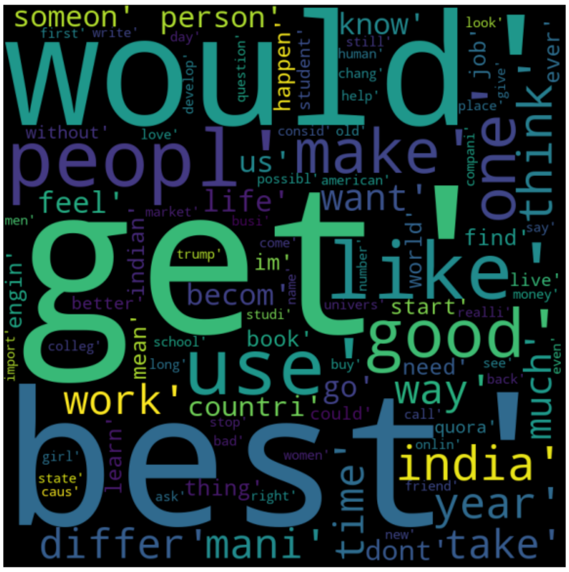
Tools

- Numpy and Pandas for data manipulation
- Scikit-learn for modeling
- Matplotlib and Seaborn for plotting
- Wordcloud and Collections for visualization
- Nltk for text processing

Communication



To exploring the balance we used catplot with our target feature. The figure depicts that the disparity is large, and the sincere questions is much more than insincere questions, so, it is imbalanced.



for identify frequent words in question text we use cloudword, so, the most frequent words are : would, get and best,…etc.