Amrita_CEN _CIQ: Classification of Insincere Questions

Chandni.M[1], Priyanga V.T[1], Premjith B[1], and Soman K.P[1]

[1] Center for Computational Engineering and Networking (CEN)
Amrita School of Engineering, Coimbatore
Amrita Vishwa Vidyapeetham, India
[2] chandnimkrishnan@gmail.com

Abstract. This paper explains about the description of the task carried out by the team Amrita_CEN_CIQ: Classification of Insincere Questions for the shared task conducted by FIRE 2019.The main objective of the shared task taken is to classify the insincere questions into six fine grained classes - Rhetorical questions, Hate speech/ inflammatory questions, Hypothetical questions, Sexually explicit/objectionable content questions, Other and Sincere/ true Information Seeking questions. The proposed system predicts the test data with an accuracy of 48.51%. The classification model used in this task is the Decision Tree Classifier. The Word embedding algorithm used for the extraction of features is Fasttext algorithm. A balanced Decision Tree is used as a classifier and proved to get better results when compared to the Random Forest Classifier with 0.52 F1-score.

Keywords: Insincere questions · fastText · Decision Tree.

CEUR-WS.org/Vol-2517/T5-2.pdf

1 Introduction

Community Question Answering (CQA) is forum which had been used by several thousands of users for seeking information and also to retrieve answers for their queries. These kind of community answer seeking websites gained popularity in the recent past and had been used by many people across the globe. However, such websites fail to provide appropriate information to the users in most of the cases because of the improper usage of forums. Quora is one such website which is facing these issues. Identification of the insincere questions posted by the users will help to resolve the above mentioned problems.

The questions posted in the forums are labelled as insincere questions by analysing the aspect of a question. This analysis and identification of the aspect

Copyright © 2019 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0). FIRE 2019, 12-15 December 2019, Kolkata, India.

of the questions are challenging and significant. Therefore, Forum for Information Retrieval Evaluation (FIRE-2019) organized a shared task on classification of Insincere Questions (CIQ). This shared task aims to categorize an insincere questions into six classes - Rhetorical questions, Hate speech/ inflammatory questions, Hypothetical questions, Sexually explicit/objectionable content questions, Other and Sincere/ true Information Seeking questions. This kind of classification not only helps the community moderators to make the website user friendly but also the users to navigate appropriate information. Thus, the CIQ had created a dataset for the classification of Insincere questions by selectively choosing Non-Information Seeking Questions (NISQ) data created by Kaggle.

We (Amrita_CEN_CIQ) developed a classification model using the Decision Tree classifier for the identification of the various aspects of insincere questions. Since, the distribution of training data is not even among all the classes, we used a weighted decision tree classifier which gave more weights to minority classes and less weight to majority classes. The sentence vectors were constructed with the fastText [8] word embedding algorithm. Our model obtained the F1-score of 0.52.

2 Description of the Task

The main objective of the shared task CIQ : Categorization of Insincere Questions was to classify the insincere questions into six different categories based on the nature of questions asked such as Rhetorical questions, Hate speech, Hypothetical classes, Sexually explicit questions, Sincere questions and other kind of questions which cannot be classified into other categories. Classification of Insincere questions(CIQ) has taken a set of questions from the Quora dataset as the training and testing data. The training set contains 900 questions across six fine grained categories of insincere questions and testing data contains 100 questions.

The number of questions in each classes are not uniform. Class "0" refers to the questions which are not sincere and contains the minimum number of questions of only 20. Whereas, Class "1" refers to the rhetorical questions which has the maximum number of questions among all other classes of classification. The unequal distribution of the data among the six different classes is one of the challenges faced during the training of the model. The statistics of the training data is given in Table 1 and a visualization of the class distribution is shown in Figure 1.

Table 1. Statistics of the training dataset

Class labels	Number of questions
0(Non-insincere questions)	20
1(Rhetorical)	488
2(Sexual Content)(Hate speech)	98
3(Hate speech)	217
4(Hypothetical)	38
5(Other)	38

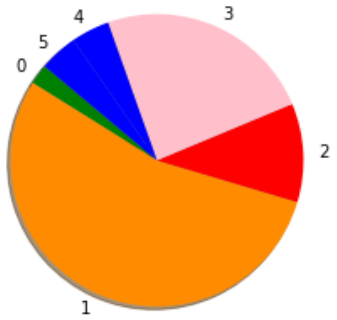
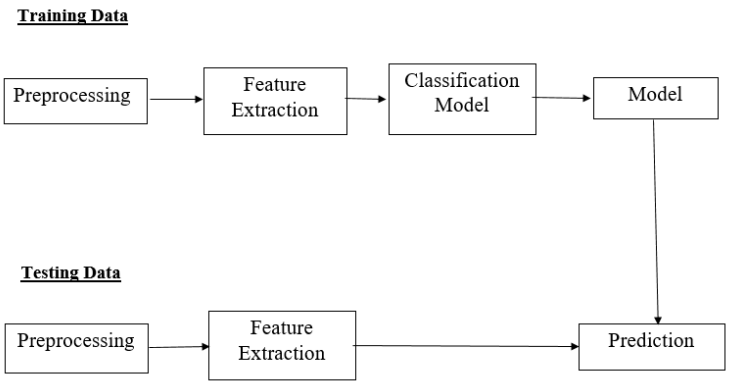


Fig. 1. Graph representing the uneven data distribution among six different classes

3 System Description

The steps involved in the developing the system submitted by Amrita_CEN_CIQ is illustrated in Figure



3.1 Preprocessing

The training and testing data are initially preprocessed to remove the non-informative content. The steps involved in the pre-processing are given as follows.

- Removal of website links and usernames
- Lower casing
- Word tokenization
- Removal of stop words
- Removal of punctuation

Each of the sentence in the corpus are tokenized into words using word_tokenize() function which are imported from the library Natural Language ToolKit(NLTK). The NLTK library also contains stopwords of 16 different languages. The stopwords in the data is also removed using NLTK library.

3.2 Feature representation

In this work, we tried fastText and Doc2vec for vectorizing the sentences with the dimension of 100. Doc2vec directly represent the sentences into a vector of dimension 100 whereas in fastText, the sentence vector was generated by taking the mean of the word vectors of constituent words. On comparison between the two algorithms used, fastText has provided a better result for the feature extraction than Doc2Vec. The parameters of the Fasttext used are tabulated in the Table 2 .

3.3 Classifier

We used weighted Decision Tree algorithm for classifying the insincere questions. The performance of the model was evaluated using 5-fold cross validation. The

Table 2. Fasttext Parameters

Parameter	Parameter Value
Algorithms used	Fasttext
Embedding size	100
Window size	3
Minimum count	1
Tokenization	Word tokenization
Epochs	50

fastText features gave the testing accuracy of 40.70 +/- 0.02% while Doc2vec features obtained 37.16 +/- 0.01%. Hence, we used word vectors generated using fastText for the submitted system. The same trend follows with the test data also. The classification accuracy of test data is 48.51% when the data were vectorized using fastText and Doc2vec gave 35.64% accuracy in identifying the different types of insincere questions. The confusion matrix for the test data is given in the following fig 3.

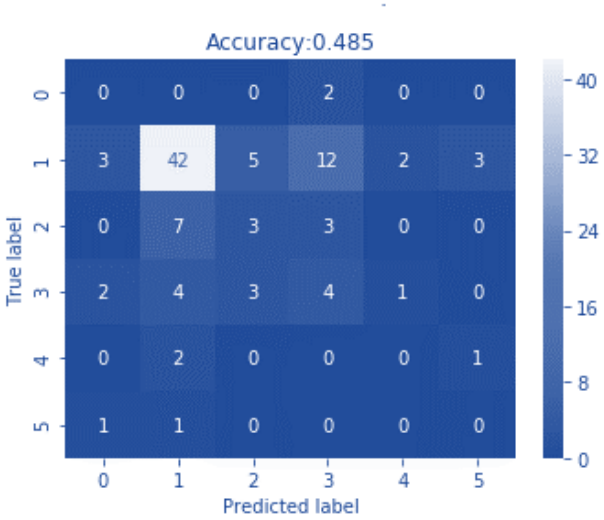


Fig. 3. Confusion matrix of the test data

The accuracy, precision and f1-score for the model is tabulated below in Table 3 with a comparison between fastText and Doc2Vec algorithms.The subset parameters which are used for building Decision Tree model and Random forest

model are tabulated in the Table 4.The advantage of a simple decision tree model that it is easy to interpret and the accuracy keeps increasing with the number of splits made in the training data. Whereas, in Random forest, the accuracy keeps increasing with the number of trees but becomes constant at a certain point of time. Unlike decision tree, it won't create highly biased model and reduces the variance. Hence, we have chosen decision tree instead of random forest as a classifier in our proposed system. The number of cross validation splits used in both cases are same which is a five-fold cross validation.

Table 3. Performance scores of decision tree classifier with fastText and Doc2vec algorithms

Metrics	fastText	Doc2Vec
Accuracy	48.51%	35.64 %
Precision	0.56	0.34
Recall	0.49	0.34
F1 score	0.52	0.33

Table 4. Parameters of the Decision Tree Classifier and Random Forest Classifier

Parameter	Parameter Value of Decision Tree	Parameter Value of Random Forest
Splitting criteria	gini	gini
Class Weight	Balanced	Balanced
Minimum samples leaf	1	1
Minimum samples split	2	2

The weighted average is used for the calculation of accuracy, precision, recall and f1-score in both the cases.

4 Conclusion

The identification of insincere questions have gained importance due to the increased number of CQA forums. Classification of insincere questions is quite easy using the text classification algorithms in NLP. This classification of insincere questions into six different classes is quite a new strategy. In this paper, we tried to implement the above mentioned classification using the fastText and decision tree algorithm for feature extraction and classification respectively. The proposed model proved to give the accuracy of 48.51% and F1-score(weighted) of 0.52 with the given test data.

References

1. Le, Quoc, and Tomas Mikolov. "Distributed representations of sentences and documents." International conference on machine learning. 2014.
2. Mikolov, Tomas, et al. "Distributed representations of words and phrases and their compositionality." Advances in neural information processing systems. 2013.
3. Joulin, Armand, et al. "Bag of tricks for efficient text classification." arXiv preprint arXiv:1607.01759 (2016).
4. Ng, Andrew. "Machine learning yearning." URL: http://www. mlyearning. org/(96) (2017).
5. Gabbard, Samuel, Jinrui Yang, and Jingshi Liu. "Quora Insincere Question Classification."
6. Craig Smith. 2019. Interesting Facts and Statistics About Quora.https://expandedramblings.com/index.php/quora-statistics
7. Kaggle Inc. 2019.Quora Insincere Questions Classification: Detect toxic content to imrpove online conversations.https://www.kaggle.com/c/quora-insincere-questions-classification
8. Bojanowski, Piotr, Edouard Grave, Armand Joulin, and Tomas Mikolov. "Enriching word vectors with subword information." Transactions of the Association for Computational Linguistics 5 (2017): 135-146.