IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Survey of various techniques for Insincere Question Classification

[1]Priyanka Pachpande, [2]Dr. Sharvari Govilkar

[1]Department of Computer Engineering,
[1]Pillai College of Engineering, Navi Mumbai, India

Abstract: With the availability of the internet anywhere and anytime people tend to move to the internet for resolution of their queries. While there are some people with genuine queries there are others who intend to misuse these CQA forums to defame someone or spread hate amongst people about an individual or group of individuals. It is crucial to identify such questions in order to stop misuse of these forums. Such insincere questions can further be classified into various categories like rhetorical questions, hypothetical questions and so on. Classification of Insincere questions (CIQ) helps to identify these insincere questions thus making CQA systems usable for genuine users. This paper presents a survey of various machine learning and deep learning techniques used for classification of insincere questions.

Index Terms **- Insincere questions, Classification of insincere questions, Random Undersampling, GloVe, BERT.**

I. INTRODUCTION

Nowadays when we face any query we turn to the internet for a solution as it provides an instant solution for any problem within seconds. For this reason many Community Question Answering (CQA) systems like Quora, Yahoo! Answers, Stack Overflow and many more sites have gained popularity in recent years. Many people use these forums to get answers to their questions as these forums give people the ability to post their queries online, and have multiple experts across the world answer them, while being able to provide their opinions or expertise to help other users, a quality that encourages more participation and consequently has led to their popularity. While many people use these forums to ask genuine questions some others use this platform to make false statements and spread hate amongst peoples. Those questions and users need to be identified in order to make these forums usable to genuine users.

II. CLASSIFICATION OF INSINCERE QUESTIONS

As the internet provides a faster way to communicate and CQA system provides a way for many people to connect and help each other more and more people are making use of CQA systems. While it is being used to benefit people there are some individuals that have bad motives use these systems to make false statements or spread hate and such questions and users need to be identified as due to such questions genuine users may not be able to get answers and some users may not even feel comfortable using this forum because of such insincere questions. For this reason it is important to classify insincere questions in CQA forums as it will help them to maintain status of these forums and genuine users can feel comfortable using such forums.

III. LITERATURE SURVEY

This section consists of the literature survey of various systems that are used in classification of insincere questions using various machine learning and deep learning algorithms.

Sourya Dipta Das, Ayan Basak, and Soumil Mandal [1], used Bidirectional LSTM - GRU model for fine grained classification of insincere questions. They Cascaded Bidirectional LSTM - GRU model with max pooling at two different points of cascading network. Two sets of concatenated word embeddings (Glove-FastText & Glove-Paragram) were used. They performed checkpoint ensembling at 31st and 32nd epochs, thus creating a total of four classification models. For checkpoint ensemble,they took the weighted average of the models at two checkpoints during the training period. Model got confused with some of the sexual content, hate speech and hypothetical questions with rhetorical questions. Model classified most of the rhetorical questions accurately because of skewed data towards rhetorical questions. Accuracy was 67.32%.

Akshaya Ranganathan, Haritha Ananthakrishnan, Thenmozhi D, Chandrabose Aravindan [2], used SGD optimizer with SVM classifier. They also used TFIDF and Count Vectorizer during the preprocessing phase. The model was unable to identify the questions which belong to the category that has less data. Accuracy was 47.52%. No pre-trained word embeddings were used.

Chandni.M, Priyanga V.T, Premjith B, and Soman K.P [3], used a weighted decision tree classifier.Distribution of training data is not even among all the classes, so they used a weighted decision tree classifier which gave more weights to minority classes and less weight to majority classes. They tried fastText and Doc2vec for vectorizing the sentences with the dimension of 100. FastText provided better results than Doc2Vec. Random forest and Weighted Decision tree models were tried and the Weighted Decision tree provides better results. 5-fold cross validation was used for performance evaluation. Accuracy of 48.51% and F1-score of 0.52 was achieved. Weighted decision tree seem to be a good option for skewed data.

Akanksha Mishra and Sukomal Pal [4], used Bidirectional LSTM followed by a dropout layer to avoid overfitting. Adam optimizer was used and to analyze overfitting validation loss was used. They tried Glove, Word2Vec, paragram and Random embeddings. Amongst them Word2Vec provided higher accuracy of 65.32%. They also applied one hot encoding for all labels.

Vandan Mujadia, Pruthwik Mishra, Dipti Misra Sharma [5], used three machine learning models Gradient boosting, Random Forest and 3-nearest neighbour classifier with majority voting. And for classification Hard Voting was used. In hard voting, the class labels are predicted based on majority voting among the participating classiers. In the case of soft voting, the voting classier picks out the maximum of the sums of the predicted probabilities computed for the constituent classiers. By performing grid-search, it was observed that combining both word unigrams and bigrams outperformed character level n-gram TF-IDF vectors as well as the combination of character and word level n-grams. They also tried LSTM, CNN + LSTM and Glove embedding + LSTM but the performance was poor compared to their machine learning model. In CQA forums like Quora, the number of spelling variations are fewer compared to social media due to character constraints. So word n-gram based TF-IDF was superior to its character counterparts. Machine Learning approach outperformed neural networks. This could be due to the higher number of parameters that deep learning approaches try to learn from a very limited amount of data. Accuracy was 62.37%.

Zhongyuan Han, Jiaming Gao, Huilin Sun, Ruifeng Liu, Chengzhe Huang, Leilei Kong, and Haoliang Qi [6], used an ensemble learning method to unite multiple classification models, including logistic regression model, support vector machine, Naive Bayes, decision tree, K-Nearest Neighbor, Random Forest. Ensemble learning uses the brew toolkit for model fusion. Brew uses the output of each classifier as a new feature value, uses a logistic regression model to learn the weights of each classifier, then outputs the classification results. TFIDF vectorizer was used. Different classifiers can learn different data features, and ensemble learning can integrate the features learned by each classification and the advantages of each classifier. The model has accuracy of 67.32% and outperforms all other models.

Samuel Gabbard, Jinrui Yang and Jingshi Liu [7], used Recurrent Neural Network (RNN) with Gated Recurrent Unit (GRU). They also implemented Naive Bayes, Logistic Regression and RNN models but found that RNN with GRU outperformed all other models. They also used word embeddings like Glove, FastText and Paragram and concluded that word embeddings outperforms TFIDF vectorizer. The model has an F1 score of 0.67. It was observed that using word embeddings in the period of data processing and using neural networks to fit the data yielded the best performance. The RNN has shown to be very useful in text classification problems because of its ability to memorize sequential inputs.

Alex Wang and Vince Ranganathan [8], used three variations of BERT and compared them with a baseline model LSTM + GRU. The three variations of BERT were Vanilla BERT, BERT + CNN, BERT + Linear. The baseline model LSTM + GRU was implemented with Glove and Paragram embedding. It was observed that the variations of BERT do not perform as well as standalone BERT, because the original BERT model is so complex and internally connected that adding additional layers dilutes the model's refined outputs. BERT model outperformed the baseline model i.e LSTM + GRU and has F1 score of 7.0. BERT uses vocabulary as a huge indicator in deciding when to flag questions, and sometimes the lack of malicious vocabulary causes BERT to miss flagging insincere questions.

Hendri Priyambowo and Mirna Adriani [9], tried multiple machine learning models like SVM, Multilayer Perceptron, Nearest Neighbor, Multinomial Naïve Bayes, Decision Tree and Random Forest along with multiple features and their combinations like Unigram, Bigram, Trigram, POSTAG, word2vec, doc2vec, Uni + Bi, Uni + Tri, Bi + Tri, Uni + Bi + Tri, Uni + POSTAG. All six models were tried with all features and combination of these features and the highest F1 score 87.81 % was achieved when using Multilayer Perceptron models with POS Tag as a feature. POS Tag features give better performance compared to other basic n-gram features. They also used Random Undersampling to handle imbalance in data by undersample the majority class.

Bishal Gaire, Bishal Rijal, Dilip Gautam, Nabin Lamichhane and Saurav Sharma [10], implemented supervised machine learning models like Multinomial Naive Bayes, KNN and Logistic Regression. Also they implemented neural networks with RNN with LSTM and GRU. It was found that the neural network has better performance than machine learning models. The neural network was implemented with the RNN model. The input layer of NN is connected to the embedding layer and output of the embedding layer is connected to the LSTM layer. The output of LSTM is connected to a bidirectional GRU followed by a dense layer. Dense layer is connected to the output layer. The F1 score of this model was 0.6913. They also used Glove and Paragram embedding with neural network.

Deepshi Mediratta and Nikhil Oswal [11], implemented Multinomial Naive Bayes, SVM, LSTM and GRU (CuDNNGRU). They also tried models without pre trained embeddings, Glove, wiki news FastText. Among these models GRU had better results followed by LSTM. GRU with Glove Embedding provided the best result. They also tried Random Undersampling and ClsuterCentroid to handle imbalance in data. Chose the Random Undersampling over ClusterCentroid because it is fast and provides better results. They have used CuDNNGRU, which is a fast GRU implementation backed by CuDNN (a GPU-accelerated library of primitives for deep neural networks). The model has accuracy of 89.49% and F1 score of 0.72.

IV. TECHNIQUES FOR CLASSIFICATION OF INSINCERE QUESTIONS

4.1 Preprocessing

Directly applying any algorithm on the dataset may affect the result as the dataset may contain some missing values or outliers. So, data preprocessing becomes essential before carrying further tasks. Data preprocessing is a task that includes preparation and transformation of data into a suitable form that will help in the task of identifying sincerity of question. Data preprocessing aims to reduce the data size, find the relation between the data, normalize and tokenize data, remove outliers and extract features for data. Data preprocessing converts the text data to analyzable and predictable form[10]. So the steps necessary to carry out under preprocessing includes:

1. removing of website links and usernames : Queries posted by users may often contain usernames and website links which are of no use in identifying the sincerity of the question. So in the preprocessing phase we remove them.
2. Word tokenization : Process of converting sentences into a chain of words so that processing word by word can be easily performed. Given a character sequence and a defined document unit, tokenization is the task of chopping it up into pieces, called tokens, perhaps at the same time throwing away certain characters, such as punctuation. We can use white space characters as delimiter for tokenization.
3. Removal of stopwords and special characters : Stop words are words which do not contain important significance to be used for identifying sincerity of the question. Few examples of stopwords are between, by, the, an, during and before.

4. correcting the misspelled words : Sometimes there are misspelled words in queries and these words could be key to identifying if a question is sincere or not. So we need to apply word correction in the preprocessing step.
5. Random Undersampling : As the dataset for CIQ is highly skewed we need to handle the imbalance otherwise the model could be highly biased towards the class with maximum data points. For this task Random Undersampling can be used. Random undersampling involves randomly selecting examples from the majority class and deleting them from the training dataset. It is a fast and easy way to balance data by randomly selecting a subset of data for the targeted class with or without replacement. A limitation of this technique is that as examples are deleted randomly, there is no way to detect or preserve good or information-rich examples from the majority class.

4.2 Word Embeddings and Features

A word embedding is a learned representation for text where words that have the same meaning have a similar representation. Word embedding is the language modeling technique in natural language processing where individual words or phrases are represented as a real-valued vector that is capable of capturing the context of the word in a document, semantic and syntactic similarities, and relation with each other word [10]. By analyzing all the papers, the below embeddings have proven to be giving better results.
1. Tfidf : TFIDF, short for Term Frequency Inverse Document Frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. The tf idf value increases proportionally to the number of times a word appears in the document and is offset by the number of documents in the corpus that contain the word, which helps to adjust for the fact that some words appear more frequently in general.
2. Glove : Glove stands for Global Vectors for word representation. The model is an unsupervised learning algorithm for obtaining vector representations for words. This is achieved by mapping words into a meaningful space where the distance between words are related to semantic similarity. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space.
3. POS tag : Part-of-Speech tagging (POS tagging), also called grammatical tagging is the process of marking up a word in a text (corpus) as corresponding to a particular part of speech, based on both its definition and its context. POS tag gives a large amount of information about a word and its neighbors.

4.3 Survey of Models and Techniques for Insincere Question Classification

There are many machine learning and deep learning approaches in survey papers. The size of the dataset available and the amount of imbalance in data are key factors in deciding which technique is effective. If the size of the dataset is less then machine learning outperforms deep learning as deep learning methods try to learn a higher number of parameters from very limited data and hence overperforms. Whereas deep learning approach outperforms machine learning when datacount is large. Below are the machine learning and deep learning approaches that have proven to be given better results for this problem statement.

4.3.1 Machine learning approach

For less data count and skewed data ensemble learning method of machine learning performs well.

4.3.1.1 Ensemble learning method of Logistic Regression, SVM, Naive Bayes, Decision Tree, KNN and Random Forest [6] :

Ensemble Learning is a way to combine multiple learning algorithms for better performance. Using this method, the model combines the advantages of all the machine learning models that are applied and then gives output. It uses the output of the first-level classifiers as the new features. Then use this new feature to train the second level meta-classifier. Hence by using this method of ensemble learning we can combine the advantages of individual machine learning models and also overcome the problems faced by those individual models.

Models / Techniques	Dataset used	Analysis / Conclusion
SGD Optimizer with SVM classifier [2]	Fine Grained Dataset provided by FIRE forum with 900 data points and 6 classification categories.	Model did not perform very well and accuracy was low. The model was unable to classify the questions which fall into the category with less training data. Accuracy - 47.52%
Weighted Decision Tree classifier [3]	Fine Grained Dataset provided by FIRE forum with 900 data points and 6 classification categories.	Using a Weighted Decision Tree was a good option as data is skewed. It performed better than Random Forest but the overall accuracy of this model is low. Accuracy - 48.52%
Gradient Boosting, Random Forest and 3 Nearest Neighbor classifier with majority voting [5]	Fine Grained Dataset provided by FIRE forum with 900 data points and 6 classification categories.	Using 3 models along with majority voting proves to be a good model. The accuracy of this model was good compared to single machine learning models. Accuracy - 62.37%
Ensemble learning method of Logistic Regression, SVM, Naive Bayes, Decision Tree, KNN and Random Forest [6]	Fine Grained Dataset provided by FIRE forum with 900 data points and 6 classification categories.	This model performs better than all other machine learning models and provides better accuracy. Different classifiers can learn different data features, and ensemble learning can integrate the features learned by each classification and the advantages of each classifier. Thus this model with an ensemble learning method outperforms other machine learning models. Accuracy - 67.32%

4.3.2 Deep learning approach

When the data count is high and data is skewed we can make use of following deep learning models as they perform well in this scenario.

4.3.2.1 BERT [8] :

BERT stands for Bidirectional Encoder Representations from Transformers. It is designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context. This pre-training step is half the reason behind BERT's success. This is because as we train a model on a large text corpus, our model starts to pick up the deeper and intimate understandings of how the language works. This knowledge is very useful in this classification task. Variations pof BERT like BERT + CNN, BERT + Linear were tried out along with plain BERT and baseline model LSTM + GRU in [8] but BERT outperformed them all. The variations of BERT do not perform as well as standalone BERT, because the original BERT model is so complex and internally connected that adding additional layers dilutes the model's refined outputs.

4.3.2.2 GRU (CuDNNGRU) [11] :

Gated Recurrent Unit (GRU) is an improved version of standard recurrent neural networks. RNN have the issue of vanishing gradient problem which is overcome by GRU. They can be trained to keep information from long ago and are able to remove information which is irrelevant to the prediction.[11] have used CuDNNGRU , which is a fast GRU implementation backed by CuDNN (a GPU-accelerated library of primitives for deep neural networks). GRU has proven to be giving better performance than LSTM and RNN models.

Models / Techniques	Dataset used	Analysis / Conclusion
GRU (CuDNNGRU) [11]	Quora dataset from Kaggle with 13,06,122 data points and 2 classification categories.	In order to handle the imbalance in data, Random undersampling is used. GRU along with Glove embedding provided the best result among machine learning and deep learning model like LSTM. Accuracy - 89.49 % and F1 score - 0.72.
Bidirectional LSTM [4]	Fine Grained Dataset provided by FIRE forum with 900 data points and 6 classification categories.	Bidirectional LSTM performed better with this dataset compared to other machine learning models but not as good as the ensemble learning model [6] using the same dataset. Accuracy - 65.32%
Bidirectional LSTM + GRU [1]	Fine Grained Dataset provided by FIRE forum with 900 data points and 6 classification categories.	Model has the same accuracy as the ensemble learning model [6] using this dataset. But this model was getting confused while classifying questions belonging to minority class as was classifying it as majority class because of skewed dataset. Accuracy - 67.32%
Multilayer Perceptron [9]	Quora dataset from Kaggle with 13,06,122 data points and 2 classification categories.	Random undersampling was used to handle imbalanced data. Multilayer perceptron with a POS tag gives better results when compared with a machine learning model. While the result was adequate but it was not as good as GRU model [11]. F1 score - 87.81 %
RNN + GRU [7]	Quora dataset from Kaggle with 13,06,122 data points and 2 classification categories.	RNN is very useful in text classification problems because of its ability to memorize sequential data. RNN with GRU provides good results because the problem of vanishing gradient in RNN is solved by GRU but still the plain GRU model outperforms the RNN + GRU model. F1 score - 0.67.
RNN + LSTM + GRU [10]	Quora dataset from Kaggle with 13,06,122 data points and 2 classification categories.	Performance of RNN + GRU model [7] is improved by using RNN + LSTM + GRU model but the improvement is less and the Plain GRU model [11] still outperforms RNN + LSTM + GRU model. F1 score - 0.6913
BERT [8]	Quora dataset from Kaggle with 13,06,122 data points and 2 classification categories.	BERT is another model apart from GRU model [11] to perform very well with this dataset. Many of the BERTs errors were actually predicted correctly which were marked incorrectly by noise in the dataset. While implementation of this model variations of BERT and LSTM + GRU model were implemented and BERT outperformed them all. F1 score - 0.70.

4.4 Evaluation Parameters

As data is skewed we cannot rely on accuracy alone, we also need to look at other parameters like Precision, recall, F1 score and confusion matrix to evaluate the performance of the model correctly.

V. CONCLUSION

The task of classification of Insincere questions can be performed using two approaches. The first one is a machine learning approach and the second is a deep learning approach. For a small dataset machine learning approach is preferred as deep learning tries to learn a higher number of parameters from very limited data and hence overperforms. And in the case of large dataset deep learning approaches outperforms machine learning ones.

Many machine learning approaches were discussed in this paper and out of those ensemble learning method stand out because of its ability to combine advantages of various machine learning algorithms and combine them to make a better model by overcoming problems faced by individual models.

We have also discussed various deep learning models used by many researchers to solve this problem but few models like BERT and GRU outperform other models like LSTM and RNN. Along with these machine learning and deep learning models, use of word embeddings after the preprocessing stage proves to be giving better performance. Few word embeddings like Glove, TFIDF and POS tag have been giving better results in this classification task.

VI. ACKNOWLEDGMENT

I take this opportunity to thank all the people who have contributed in some way to the work described in this paper. I am most grateful to my project guide for providing timely advice and for her invaluable support. I express my thanks to the Head of Computer Department and to the Principal of Pillai College of Engineering (PCE), New Panvel for extending their support.

REFERENCES

[1] Sourya Dipta Das, Ayan Basak, and Soumil Mandal, "Fine Grained Insincere Questions Classification using Ensembles of Bidirectional LSTM-GRU Model", FIRE 2019 - Forum for Information Retrieval, December 2019

[2] Akshaya Ranganathan, Haritha Ananthakrishnan, Thenmozhi D, Chandrabose Aravindan, "Classification of Insincere Questions using SGD Optimization and SVM Classifiers", FIRE 2019 - Forum for Information Retrieval, December 2019

[3] Chandni.M, Priyanga V.T, Premjith B, and Soman K.P, "Amrita CEN CIQ: Classification of Insincere Questions", FIRE 2019 - Forum for Information Retrieval, December 2019

[4] Akanksha Mishra and Sukomal Pal, "IIT-BHU at CIQ 2019 : Classification of Insincere Questions", FIRE 2019 - Forum for Information Retrieval, December 2019

[5] Vandan Mujadia, Pruthwik Mishra, Dipti Misra Sharma, "Classification of Insincere Questions with ML and Neural Approaches", FIRE 2019 - Forum for Information Retrieval, December 2019

[6] Zhongyuan Han, Jiaming Gao, Huilin Sun, Ruifeng Liu, Chengzhe Huang, Leilei Kong, and Haoliang Qi, "An Ensemble Learning-based model for Classification of Insincere Question", FIRE 2019 - Forum for Information Retrieval, December 2019

[7] Samuel Gabbard, Jinrui Yang and Jingshi Liu, "Quora Insincere Question Classification", [Online] Available : https://www.semanticscholar.org/paper/Quora-InsincereQuestion-Classification-Gabbard-Yang/f469033055a13eb749808db61c120c12a8cb2bf3, 2019

[8] Alex Wang and Vince Ranganathan, "Is this question sincere? Identifying insincere questions on Quora using BERT and variations", [Online] Available: http://web.stanford.edu/class/cs224n/reports/custom/15763730.pdf, 2019

[9] Hendri Priyambowo and Mirna Adriani, "Insincere Question Classification on Question Answering Forum", 2019 International Conference on Electrical Engineering and Informatics (ICEEI), DOI: 10.1109/ICEEI47359.2019.8988798, July 2019

[10] Bishal Gaire, Bishal Rijal, Dilip Gautam, Nabin Lamichhane and Saurav Sharma, "Insincere Question Classification Using Deep Learning", International Journal of Scientific & Engineering Research Volume 10, Issue 7, DOI: 10.13140/RG.2.2.30392.49925, July 2019

[11] Deepshi Mediratta and Nikhil Oswal, "Detect Toxic Content to Improve Online Conversations", [Online] Available : https://arxiv.org/abs/1911.01217v1 , October 2019