Insincere Question Classification Using Deep Learning

Bishal Gaire[1], Bishal Rijal[2], Dilip Gautam[3], Nabin Lamichhane[4], Saurav Sharma[5]

[1,2,3,4,5] Department of Electronics and Computer Engineering,
Tribhuvan University, Institute of Engineering, Paschimanchal Campus,
Lamachaur-16, Pokhara, Nepal

Email: [1]bsalgaire360@gmail.com [2]bishalrijal110@gmail.com [3]hiddendreamz7@gmail.com
[4]babunabin@wrc.edu.np [5]sauravsapkotasharma@gmail.com

Abstract—Thousands of Questions are asked in QA forms every day and manual classification of them as 'sincere' and 'insincere' is unrealistic. Insincere questions affect the user experience across the platform and are a severe concern. Thus, to dilute such questions we implement a machine learning model for such Question classification. Dataset was provided by Quora through the website Kaggle and it consists of a training set of over 1.3 Million labeled examples and around 300 thousand of unlabeled examples as a test set. We implemented artificial recurrent neural network (RNN) architecture such as a Long Short-Term Memory (LSTM) unit and a recently proposed gated recurrent unit (GRU). Only the training data set was used in our work. We have used 10% of the data set for cross-validation, and we found an F1 score 0.6913 when the threshold was set to 0.35.

Index Terms—Insincere Question Classification, Sentimental Analysis, Natural Language Processing, Deep Learning.

IJSER

1 INTRODUCTION

Quora is a question-answer platform where users submit questions and seek for the relevant answers or opinions. On Quora, people can ask questions and join with others who contribute genuine quality answers and opinions. So, the quality of questions becomes an essential part of the community. The majority of users on Quora are well-intentioned and ask questions they're genuinely interested in but in few cases, someone will ask a question in a deliberately provocative way, where the wording is designed to make its own statement. This may include prejudicial framing or calls to confirm hateful stereotypes. These questions are harmful to the Quora community, and Quora remove or hide them whenever it becomes aware of them. A key challenge is to weed out insincere questions -- those founded upon false premises, for that intends to make a statement rather than look for helpful answers. Questions based on false or absurd assumption, intentionally using not appropriate sexual content, intended to make a statement about a certain group of people. Looking for confirmation to stereotypes about social groups can be characterized as insincere [1].

Eventually, the major objective of this research is to predict whether the question asked by the user is sincere or insincere.

2 LITERATURE REVIEW

To address this problem Quora has employed both machine learning and manual review.

As pointed out by Yoon Kim on paper entitled 'Convolutional Neural Networks for Sentence Classification' has performed a series of experiments with CNN on pre-trained word-vector for sentence classification. According to his paper, a little hyper parameters turning and static vectors in a simple model (CNN-statics) perform remarkably well, giving competitive results against the more sophisticated deep learning model that utilizes complex pooling schemes. This CNN model improves upon the state of the art on 4 out of 7 tasks which include sentimental analysis and question classification [2].

According to a researcher of CMU (Carnegie Mellon University) and Microsoft Research jointly wrote the paper entitled 'Hierarchical Attention Networks for Documents Classification' in 2016. An experiment was conducted on six large scale Datasets. Results of the experiments showed that for smaller datasets such as Yelp 2013 and IMDB this model outperforms the previous best baseline methods by 3.1% and 4.1% respectively. For large datasets, this model outperforms the previous best model by 3.2%,3.4%,4.6%and 6.1% on Yelp 2014, Yelp 2015, Yahoo Answers and Amazon Reviews [3].

Prudhvi Raj, Dachapally and Srikanth Ramanam presented the paper entitled 'In-Depth Question Classification Using Convolutional Neural Network. According to their paper typically CNN is used for image classification. CNN for NLP is not used often and is completely intuitive. They used two-tier CNN that classifies questions into their main and sub-categories. The architecture consists of one Convolutional layer that learns several filters for given heights (Bi-grams to Pent-grams), after that 2-max-pooling layer that accumulates more information from the convolution layer. All the max-pooled layers were merged to form a 2-fully connected layer with node 128 and 64.

The data used for training was questioned classification dataset by the University of Illinois, Urbana Champaign. While testing their model, it was found that 90.43% main category accuracy and 76.52% subcategory accuracy for the Quora dataset which was manually collected. For TREC 93.4% was the main category accuracy and 87.4% subcategory accuracy [4].

Abdalraouf Hassana and Ausif Mahmood at the University of Bridgeport has done research on 'Deep Learning for Sentence Classification'. As observed by the paper, most of the machine learning algorithm requires the input to be denoted as a fixed-length feature. A bag of words is a popular fixed-length feature. It is simple but is limited in many tasks. They ignore the semantics of word and loss ordering of words. So, Long Short-Term Memory (LSTM) is used over a pre-trained word vector to capture semantic and syntactic information.

In the process of trying to predict whether a question is insincere, they used pre-trained word vector, which was trained on 100 billions of words of Google News. The use of a pre-trained word vector offers several advantages. A similar word is clustered together. LSTM is used to avoid the problem of vanishing gradient.

In their experiment, they used two datasets for sentiment analysis: Stanford Large Movie Review Dataset IMDB and Stanford Sentiment Treebank (SSTb). The training was done through stochastic gradient descent over shuffled mini-batches. The size of the hidden state was to be 128 and the mini-batch size was 64. Dropout was set to 0.5 .10% of training data was taken for validation.

Their model provides a 14.3% error rate for SSTb and an 11.3% error rate for IMDB [5].

Ashwin Dhakal and his co-authors, in their paper "Exploring Deep Learning in Semantic Question Matching" has implemented Artificial Neural Network approach to predict the semantic coincidence between the question pairs, extracting highly dominant features and hence, determining the probability of question being duplicate in Quora. In their research works, the words and phrases are mapped into vectors of real numbers followed by feature engineering, which includes NLTK mathematics, Fuzzy wuzzy features, and Word mover distances combined with vector distances [6].

Hence, the research has discussed and following the architecture used by the Quora itself along with the knowledge of natural language processing and machine learning.

3 METHODOLOGY

We have performed our research by performing out the below sections.

3.1 Data Extraction

Quora provided out a dataset which contains 1303122 numbers of questions. The dataset contains three labels 'qid', 'question_text' and 'target' which is a binary value and is labeled as 1 for an insincere question or otherwise. Similarly, the Quora has also provided out a test dataset that contains 375806 of test data which contains only two labels 'qid' 'question_text'. We have only used training data for our experiment. Head of the training data is shown over Fig. 1.:

	qid	question_text	target
0	00002165364db923c7e6	How did Quebec nationalists see their province...	0
1	000032939017120e6e44	Do you have an adopted dog, how would you enco...	0
2	0000412ca6e4628ce2cf	Why does velocity affect time? Does velocity a...	0
3	000042bf85aa498cd78e	How did Otto von Guericke used the Magdeburg h...	0
4	0000455dfa3e01eae3af	Can I convert montra helicon D to a mountain b...	0

Fig. 1. Head of dataset using pandas

3.2 Dataset Description

Analysis of the dataset was done by plotting the graph and using the pandas. We found that there were 1225312 numbers of questions that were sincere, label by 0 and 80810 numbers of the question were insincere, label by 1.

Bar plot of dataset:

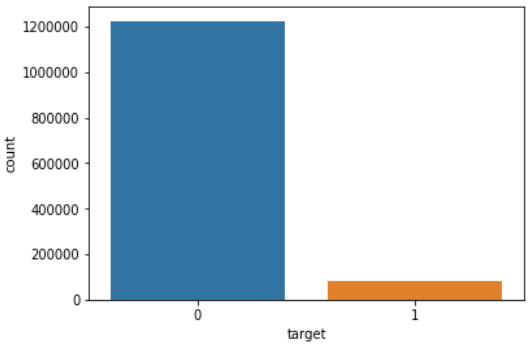

Fig. 2. number of sincere and insincere question .Here 1 represent insincere and 0 represent sincere questions.

This shows 93.81% is sincere question and 6.18% insincere, this shows that the dataset contains class imbalance problem.

IJSER

3.3 Data 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. Data preprocessing aims to reduce the data size, find the relation between the data, normalize data, remove outliers and extract features for data [7]. Data preprocessing converts the text data to analyze and predictable form. So the steps necessary to carry out under preprocessing includes:

1. removing of special characters
2. removing space
3. replacing the number by special characters
4. cleaning the misspelled words

3.4 Word embedding

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. We have implemented the embedding layer as a concatenation of two pre-trained word embeddings, GloVe, paragram, jointly with neural Network.

GloVe:

The GloVe model is a log-bilinear model with weighted least-squares objectives. Training of this model is based on the simple observation that the ratio of the word-word co-occurrence probabilities. These ratios encode some form of meaning; extract the relationship among the words [8].

Paragram:

Paragram is the compositional model. Paragram encode arbitrary word sequences into a vector as Glove. Training of this model is based on those sequences with similar meaning have high similarities [9].

3.5 Supervised machine learning model:

We have implemented three supervised learning algorithm and we found the following results:

SN	Supervised learning Algorithm	Encoder	F1 Score
1	Multinomial Naive Bayes	TF-IDF	0.1340
2	K-nearest	TF-IDF	0.211
3	Logistic Regression	TF-IDF	0.540.

Table 1 :F1 score of supervised learning algorithm

3.6 Design of neural network

We have used RNN for making model. RNN is a type of neural network in which the output from the previous step is fed as input to the current step. Our neural network consists of the input layer, 5 hidden layers, and 1 output layer. The input layer consists of 65 nodes. This input layer is connected to the embedding layer. This embedding layer is used for creating the vector representation of words. Weights of the embedding layer are initialized by the concatenation of third party embeddings (GloVe and paragram). So the embedding matrix is of 600 dimensions. The embedding layer has a spital dropout value of 0.4. The output of the embedding layer is connected to the input of the LSTM layer. There are 150 bidirectional LSTM cells in the LSTM layer. Kernel's initializer for this layer is "glorot uniform" and the recurrent initializer is orthogonal. The output of the LSTM is connected to the bidirectional GRU. There are 150 cells in the GRU layer. Attention later is attached to the GRU followed by the dense layer with 36 nodes with dropout 0.1. ReLU function i.e. **$A(x) = max(0, x)$** was used as an activation function in this layer. A dense layer is connected to the output layer. Here Sigmoid function i.e **$f(x) = 1/(1+e^{-x})$** was used as an activation function.

3.7 Training of neural network

The whole dataset was splited to train set and validation set.10% of the data set was used for validation. Sequence padding was done to make all the text of equal length. Pre-zero padding was done since LSTM and GRU were used [10]. Since the classification is binary classification so Binary cross-entropy was used as loss function [11]. A sequence of length 65 was the input to the model.

The batch size was 2500 and 7 epochs was used for training the model. Adamax optimizer with a learning rate of 0.003 and decay 0.00002 was used. 0.35 was used as a threshold value. During training the model we found that at threshold 0.35 the F1 score was 0.6913.

Loss during training and validation test is shown below:

IJSER

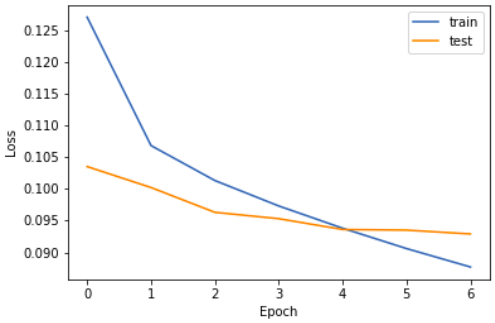

Fig. 3. Loss during training and validation at different epoch

F1 score was determined by finding the threshold. Below figure shows the plot of F1 score and threshold.

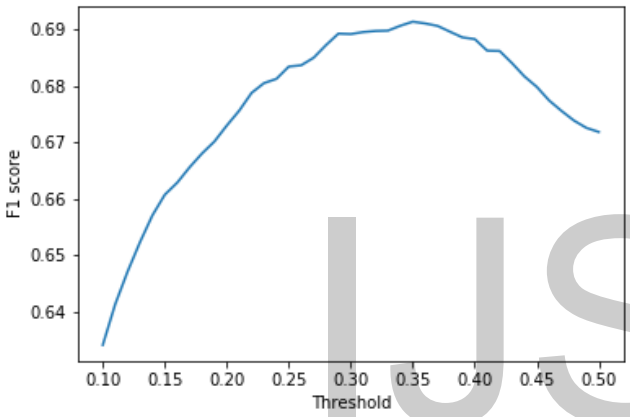

Fig. 4. F1 score at different threshold value

The Confusion matrix table for the sincere and insincere question is shown in the figure below:

119699 (True Negative)	3049 (False positive)
2099 (False Negative)	5766 (True Positive)

Table 2 : Corresponding table of confusion

From above table we calculate F1 score as:

Precision *=True Positive/ (True Positive + False Positive)*

Recall *= True Positive/ (True Positive + False Negative)*

F1 *= 2 * (Precision* Recall/ Precision +Recall)*

Eventually we obtained the following results:

Precision = 5766/ (5766+3049)
=0.65411
Recall = 5766/ (5766+2099)
= 0.73312
F1 = 0.6913

4 Results

We obtained different results using different supervised machine learning algorithm as shown in Table 1. Among them Logistic Regression gave the best result. When implementing on RNN, the best result of our model was noted to be 0.6913.

5 Discussions and Conclusion

So, this research work is based on Natural Language Processing to address the problem of handling toxic content in Q&A forums by applying Deep Learning to classify whether question is sincere or not.

6 References

[1] "Quora Insincere Questions Classification." [Online]. Available: https://kaggle.com/c/quora-insincere-questions-classification. [Accessed: 29-Jul-2019].
[2] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014.
[3] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical Attention Networks for Document Classification," *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2016.
[4] Dachapally Prudhvi Raj, "In-depth Question classification using Convolutional Neural Networks," *arXiv preprint arXiv:1804.00968*, 2018.
[5] A. Hassan and A. Mahmood, "Deep learning for sentence classification," *2017 IEEE Long Island Systems, Applications and Technology Conference (LISAT)*. 2017.
[6] A. Dhakal, A. Poudel, S. Pandey, S. Gaire, and H. P. Baral, "Exploring Deep Learning in Semantic Question Matching," presented at the 2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS), Kathmandu, 2018, pp. 86–91. doi: 10.1109/CCCS.2018.8586832
[7] S. Alasadi, "Review of Data Preprocessing Techniques in Data Mining," *Journal of Engineering and Applied Sciences*, vol. 12, pp. 4102–4107, 2017.
[8] J. Pennington, R. Socher, and C. Manning, "Glove: Global Vectors for Word Representation," *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2014.
[9] Wieting, John Bansal, Mohit Gimpel, Kevin Livescu, Karen, "Towards universal paraphrastic sentence embeddings," *arXiv preprint arXiv:1511.08198*, 2015.
[10] M. Dwarampudi and N. V. Subba Reddy, "Effects of padding on LSTMs and CNNs," *arXiv [cs.LG]*, 18-Mar-2019.
[11] J. Brownlee, "Loss and Loss Functions for Training Deep Learning Neural Networks," *machinelearningmastery.com*, 28-Jan-2019. [Online]. Available: https://machinelearningmastery.com/loss-and-loss-functions-for-training-deep-learning-neural-networks/.

IJSER