National College of Ireland

To what extent NLP with RNN and Transformer Based Deep Neural Network can be used to classify Insincere questions on Quora.

MSc. Research Project
Data Analytics

Rohit Kumar Shrivas
Student ID: x19226403

School of Computing
National College of Ireland

Supervisor: Hicham Rifai

National College of Ireland
Project Submission Sheet
School of Computing

National College of Ireland

Student Name:	Rohit Kumar Shrivas
Student ID:	x19226403
Programme:	Data Analytics
Year:	2021
Module:	MSc. Research Project
Supervisor:	Hicham Rifai
Submission Due Date:	16/08/2021
Project Title:	To what extent NLP with RNN and Transformer Based Deep Neural Network can be used to classify Insincere questions on Quora.
Word Count:	4976
Page Count:	21

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Rohit Kumar Shrivas
Date:	23rd September 2021

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

To what extent NLP with RNN and Transformer Based Deep Neural Network can be used to classify Insincere questions on Quora.

Rohit Kumar Shrivas
x19226403

Abstract

This research project adopts a novel approach of utilizing BERT based models for question classification application of NLP. The performance of these models are then evaluated using F1 score and compared with the performance of RNN based Deep Learning models with and without Attention layer. The DistilBERT model outperformed every model implemented in the research and achieved an F1 score of 94.87% which is 7.06% improvement over previous studies. Among the RNN based Deep Learning models, Model-6 (Bi-directional GRU + Bi-directional LSTM + Attention layer) performed the best and achieved an F1 score of 63.53%.

1 Introduction

The Internet has rapidly changed the lifestyle of people recently. It had bridged the gap between physical and virtual reality. This is particularly true for shopping as it has become common to perform shopping online especially during the Covid-19 pandemic. As part of the online shopping experience, the general public consults the online community question and answers to review some of the comments about a particular product. Consequently, the online community question and answer forums such as Quora and Yahoo Answers have boomed and are some of the most used public websites. These forums have seen a huge boom in usage in the recent decade and the content is getting richer in people's opinions, personal feelings, emotions etc. Simultaneously, these forums are also prone to an excessive amount of malignant and misleading content and it has become easier to spread destructive opinions through these platforms utilizing the richer content.

Natural Language Processing (NLP) and Natural Language Understanding (NLU) have been widely used in user interface chatbots such as Amazon's Alexa, Apple's Siri etc. which essentially extracts the key intentions of the user from the natural language and communicate with them in the same to establish an active human-computer dialogue(1)(2). Text Classification is a very elementary yet dynamic application of NLP and has shown decent promise when used in conjunction with machine learning (ML) algorithms. The combination is used to achieve a wide variety of objectives including, sentiment analysis, subjectivity and question classification etc.(3)(4)(5)(6)(7).

In recent years, Deep Neural Network (DNN) based models have advanced heavily and are providing better results when compared to conventional machine learning models.

Recurrent Neural Network (RNN) and Transformer based DNN such as BERT in conjunction with NLP are proving their worth when dealing with large data and are slowly replacing the conventional ML algorithms.

1.1 Motivation

The open nature of these online community Q&A forums has been a major reason for their success. However, the same has made them vulnerable to hateful, destructive and malignant content. While most of the users utilize these online community forums for a genuine purpose, some use them to spread destructive and hateful content (8)(9)(10). A recent example of this could be seen through the excessive amount of racist and hateful comments made over players of colour playing for England's football team after the Euro cup final of 2021, and against Sir Lewis Hamilton after the high-speed crash in Formula 1 Silverstone Grand Prix of 2021 involving current championship rival drivers Max Verstappen and Sir Lewis Hamilton who is the only F1 racing driver of colour in the entire history of Formula 1 racing. This insincere content that is uploaded into these websites on a daily basis has to be identified and removed to make these websites constructive and ensure a safe surfing experience.

1.2 Research Question

To what extent NLP with RNN and Transformer Based Deep Neural Network can be used to classify Insincere questions on Quora.

1.3 Structure of Project

The structure of this research report is as follows:

Chapter 2: Literature Review This section of the report sheds light on the previous studies carried out in the field of Natural Language Processing (NLP) applications and on the usage of similar models or architecture that have been implemented in this project. This sections discusses the results achieved by them alongside the advantage and disadvantage of them.

Chapter 3: Methodology This section of this report highlights the methodological approach adopted for implementation of this project. Also, the details about the data including collection of it from source to exploration is discussed in this section.

Chapter 4: Design Specification This section of this report explains the framework adopted for the implementation of this research project and briefly discusses various stages of it.

Chapter 5: Implementation This section of this report sheds light on the architecture and working of different RNN based deep learning models and Transformers based models implemented in this research project.

Chapter 6: Evaluation This section of the report introduces the evaluation metrics utilized in measuring and comparing the performance of each implemented model. The results obtained by the models are also discussed in detail.

Chapter 7: Conclusion and Future Work This section of the report summarizes and concludes the findings of the research project. It also points out ways in which this research could be carried forward and improved.

2 Literature Review

Conventional machine learning algorithms have always been the popular go-to method for natural language processing applications until recently when deep learning based Neural Network algorithms started to show promising results. The deep learning based neural network algorithms were developed especially to deal with the extensively large amount of data while keeping a high rate of precision in results. Since the data generated by these online community Q&A forums is colossal and is increasing day by day, the researchers could not resist implementing these algorithms to NLP applications such as question classification, sentiment analysis, text classification, translation between languages, text summarizing etc.

Previous studies carried out in the question classification and text classification applications of NLP has been discussed in section 2.1 and section 2.2 respectively. Meanwhile, section 2.3 sheds light on relevant studies which made use of transformer based model approach for different NLP applications.

2.1 Question Classification

The raw data usually consists of a significant amount of noise and these need to be removed. The key elements holding essential messages are identified from training data through the bag of words approach. NLP methods like Tokenisation are the example of such methods. TF-IDF and PCA are some other methods that fall under the umbrella of NLP which are utilized to reduce the dimensionality of the training data by retaining only the essential elements and attributes (11). Authors of (12) attempted to classify insincere questions and in doing so compared the performance of Deep learning based Multilayer Perceptron (MLP) algorithm with multiple conventional machine learning algorithms such as Random Forest, Decision Tree, Support Vector Machine (SVM), K-nearest Neighbour and Multinomial Naïve Bayes. The authors also recorded and highlighted the influence of multiple feature selection techniques such as unigram, bi-gram, n-gram, POSTAG etc. both separately on each algorithm and in combination of several with them as well. The research attained the best result with a model containing MLP algorithm in conjunction with POSTAG feature selection technique which achieved an F1 score of 87.81%. Meanwhile, in general the conjunction of Unigram and POSTAG feature selection techniques with both deep learning based algorithm and conventional machine learning algorithm provided the most consistent results. Authors of (13) proposed a novel hybrid deep learning model consisting of a self-attention layer with Bi-LSTM layer and CNN layers, for the purpose of question classification. The proposed model comprises of word2vec and Chinese synonym dictionary followed by a self-attention layer which identifies the potential linguistic relation between vectors. These are then fed to the Bi-LSTM layer which computes the semantic representation from it and feeds it to the next CNN layer having max-pooling. The CNN layer extracts linguistic features and feeds them to the next fully connected layer followed by a SoftMax layer. The SoftMax layer situated at the end provides the final classification result. The authors have compared the performance of their proposed hybrid model with various conventional machine learning algorithms and deep learning algorithms such as Support vector machine (SVM) and Convolutional Neural Network (CNN) respectively alongside a few more. The research attained the best result from the proposed hybrid model followed by the CNN algorithm and SVM algorithm respectively. It was observed that the proposed model

achieved an accuracy of 7.57% and 9.37% higher than that of CNN and SVM algorithms respectively. A psychological study reports a positive correlation between profanity and dishonesty (14). Authors of (15), inspired by the findings of the study, attempted to classify insincere questions using a novel model which incorporates a profanity-based classifier with conventional machine learning algorithms and Neural Network based deep learning model. The research found a significant rise in performance of both conventional machine learning algorithms and Neural Network based deep learning algorithms with the integration of profanity-based classifier. The Neural Network based deep learning algorithms performed better than the conventional machine learning algorithms in both cases i.e., with and without integration of profanity-based classifier. However, since the classifier utilizes the bag of word approach which does not essentially consider the context of words in the sentence, the feasibility of this result is questionable and cannot be relied upon as it may provide a false positive output.

2.2 Text Classification

Authors of (3) attempted eight different NLP based applications such as Sentiment analysis, movie review, opinion polarity detection, question classification etc. using a novel Stacked Residual Cross-Layer Attention (SRCLA) model. The stacked structure of the model was responsible for filtering out the key features from the data while the cross-layered attention was responsible for refining those features. This research found that the proposed model was performing better than Deep Recursive Neural Network (DRNN) model (16), Tree structured Long-Short Term Memory (Tree-LSTM) model, Tree Based Convolutional Neural Network (TBCNN) model (17), Linguistic Regularized Long-short Term Memory (LR-BiLSTM) model (18) etc. To further validate the linguistic feature filtering process, the authors also added a character-level embedding through CNN as input at the initial layer to SRCLA and named the model as Char-CNN-SRCLA which showed marginal improvements over the results provided by SRCLA. While the SRCLA outperformed all the competitor models in seven out of eight classification applications including sentiment analysis, movie review etc. and achieved the highest accuracy, the Self-adaptive hierarchical sentence (AdaSent) model provided the highest accuracy in opinion polarity detection.

2.3 Transformer based NLP application approaches

Authors of (19) compared the performance of various versions of Bidirectional Encoder Representations from Transformers (BERT) pre-trained models (20) such as Robustly optimized BERT approach (RoBERTa), DistilBERT (A Distilled version of Bidirectional Encoder Representations from Transformers) (21) and Generalized Auto-regression pre-training for Language Understanding (XLNET) while implementing them for text-based emotion recognition using the ISEAR dataset which consists of seven different types of emotions such as anger, joy, guilt, fear etc. The research observed that the RoBERTa attained the highest accuracy of 74.31% in doing so, which was closely followed by XLNET, BERT and DistilBERT with the accuracy of 72.99%, 70.09% and 66.94% respectively. It was also observed that DistilBERT required the least computational resource and was the fastest among the four while XLNET demanded the most amount of computational resources and was the slowest among the four. The authors of (22) carried out a study to verify facts through the Fact Extraction and Verification (FEVER) dataset (23) and

utilized BERT (20), RoBERTa and Electra models in doing so. The study compares the performance of these models based on accuracy and F1 score. The research found that the RoBERTa model outperformed both BERT and Electra models. RoBERTa achieved an F1 score of 95.3% closely followed by BERT which achieved an F1 score of 94.3%. Electra however, performed very poorly and managed to achieve an F1 score of 66% only. It was also observed that the BERT model took a significant amount of extra computing time when compared to the RoBERTa model. The authors of (24) utilized the DistilBERT model (21) to classify commit messages into categories of Corrective, Adaptive and Perfective. The performance of the model is evaluated using the F1 score measure and is compared to the results of previously carried out studies. The proposed model significantly outperformed the results of previous studies and achieved an F1 score of 87% and hamming loss of 11%. However, the feasibility of this result could not be relied upon as each dataset has different rules set up to assemble commit messages and the classifiers might encounter ambiguity due to this same reason. The authors of (25) proposed the BART-TextRank model which is based on the original BART model for the purpose of text summarization application. The results of this proposed model are then compared to that of the original single-BART model using the evaluation metrics ROUGE. The proposed model outperformed the single-BART model and improved the result by 1.5%.

3 Methodology

The methodology adopted for the sake of this research project is the Cross-Industry Standard Process for Data Mining (CRISP-DM) which consists of six procedural stages positioned circularly as shown in Figure 1. It is widely preferred as it provides a uniform framework for guidelines, managing projects etc. Also, its stages are allowed to be processed in different orders to meet the requirements of projects and allows tracking a previous task and repetition of actions (26).

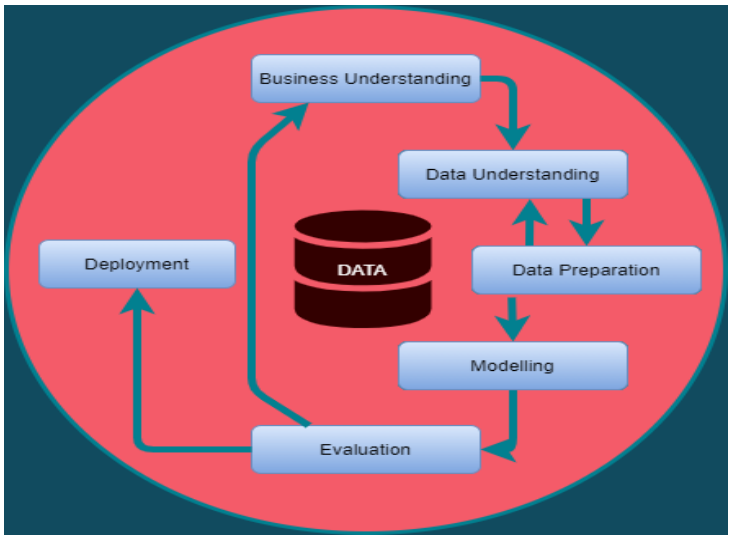


Figure 1: CRISP-DM Methodology

3.1 Business Understanding

Having an understanding of the needs of a business is imperative to CRISP-DM methodology, and hence is given utmost importance by keeping it as the first stage. The need must be well understood before converting it into an analytical problem which will lay down the foundation towards the development of a strategy for uncovering the solution. If this stage is not given the importance it demands, all the time and resources might be utilized to solve a wrong problem statement. For this study, a public dataset containing various questions asked on an online community Q&A forum named Quora has been utilized to identify insincere questions. This research study will assist the websites in identifying/classifying the insincere content which could then be removed to ensure a safe surfing experience for its users.

3.2 Data Understanding

This phase of the CRISP-DM methodology deals with the collection, understanding and exploration of the data. This research utilizes a competition dataset consisting of a question set from Quora asked by users on their website which is available on Kaggle, a public dataset platform. The dataset consists of 3 columns i.e., 'qid' which contains a unique alphanumeric id to identify each question text, 'question_text' which contains the questions asked by the users on the online community Q&A forum of Quora and 'target' which consists of the two numbers '0' to identify respective question as sincere and '1' to identify respective question as insincere. The dataset contains 1,306,122 rows each containing a question asked by the users. The dataset has been provided by Quora itself and has categorised a question insincere if it falls under either one or more of the below-mentioned categories: • Using sexual content including incest, paedophilia etc. for bogus reasons. • Has exaggerated tone to prove a point against and undermine a certain group of people. • Contains discriminatory or disparaging content against a specific group of people. • Contains false facts based on irrational assumptions.

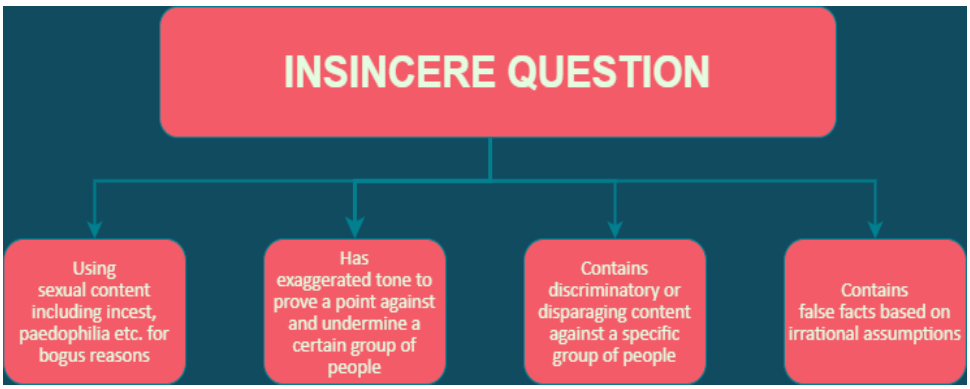


Figure 2: Insincere question categories

3.3 Data Preparation

At this stage of the CRISP-DM methodology, the data is prepared to develop the final version which will be utilized for modelling. This process includes all the procedures required to obtain a clean and organized dataset as per the requirement of the project

modelling tools. In this research, the following operations were performed to obtain a final version of the raw dataset downloaded from Kaggle:

3.3.1 Data Collection

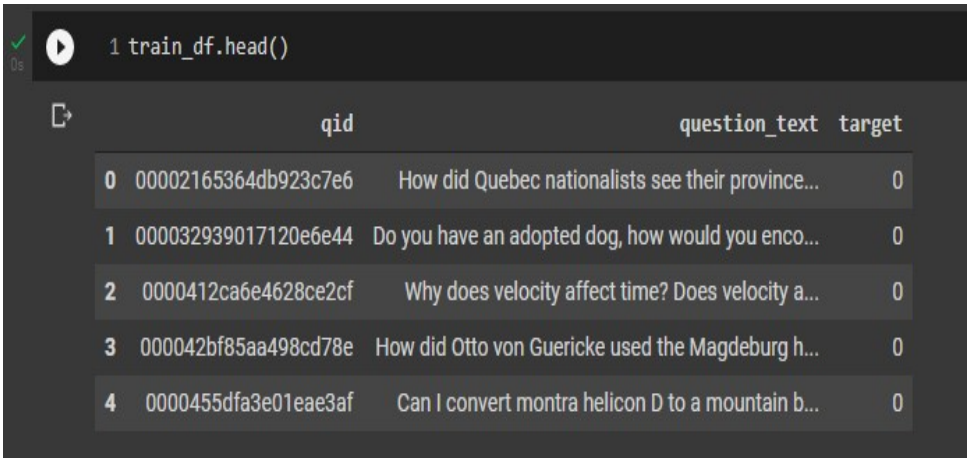
The dataset has been downloaded and stored in google drive `https://drive.google.com/file/d/1ktAT2mqPsRe2WmImqMIfFJkIvUwe_bMN/view?usp=sharing` and has been fetched to the Google Colab Notebook using the section of code shown in Figure 3. The code allows the Google Cloud SDK to link to your drive account via a one-time verification alphanumeric code. The sharable link to the file is then utilized to locate the dataset in your drive and the same could be uploaded to a variable in Colab Notebook. There are 2 more popular ways to upload a .csv file into Google Colab Notebook. However, considering the size of the file, this was the fastest way and was hence preferred over the others.

```
[ ]  1 # Authenticate and create the PyDrive client.
     2 auth.authenticate_user()
     3 gauth = GoogleAuth()
     4 gauth.credentials = GoogleCredentials.get_application_default()
     5 drive = GoogleDrive(gauth)

[ ]  1 link_train = 'https://drive.google.com/file/d/1ktAT2mqPsRe2WmImqMIfFJkIvUwe_bMN/view?usp=sharing' # The shareable link

[ ]  1 train_id = '1ktAT2mqPsRe2WmImqMIfFJkIvUwe_bMN'

[ ]  1 downloaded = drive.CreateFile({'id':train_id})
     2 downloaded.GetContentFile('train.csv')
     3 train_df = pd.read_csv('train.csv')
```

Figure 3: Uploading the data into Google Colab from Drive

3.3.2 Data Pre-processing

The dataset uploaded into the notebook consists of 3 columns and 1306122 rows containing the questions asked by users on Quora. The following procedures were operated on the dataset to prepare it for the modelling process.



```
1 train_df.head()
```

	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

Figure 4: A look at the data

(a) *Splitting the data into train and validation sets*: The data was then split into training and validation sets (into the ratio of) using the train_test_split function from the sklearn library.

(b) *Removing stop-words, prepositions, conjunctions and lemmatizing the question text*: At this stage the data is written in natural English language which consists of prepositions, conjunctions, stop-words and contractions of words in a significant amount to affect the performance of the implemented models. These are then removed to make the data cleaner and more meaningful which is then ready to be fed to applied models.

(c) *Filling N/A values*: All the missing values in the question text attribute in both the training dataset and validation dataset were then filled with '_NA_'.

(d) *Tokenizing the sentences*: Tokenizing is the process of splitting the natural language sentences, in this case, questions; into smaller units called tokens. These tokens help in identifying the weight of the word in each sentence and hence conjunctions and others could be identified which in essence is equivalent to identifying the most meaningful word of a sentence and therefore understanding the meaning of the whole sentence. This operation is carried out using the Tokenizer function of the keras library as shown in Figure 5. Also, the code used to tokenize questions for transformers based models is shown in Figure 6.

```
13 ## Tokenize the sentences
14 tokenizer = Tokenizer(num_words=max_features)
15 tokenizer.fit_on_texts(list(train_X))
16 train_X = tokenizer.texts_to_sequences(train_X)
17 validation_X = tokenizer.texts_to_sequences(validation_X)
18
```

Figure 5: Tokenizing the sentences

```
[ ]  1 #Tokenizing the samples
     2
     3 train_x = fast_encode(train_set['question_text'].astype(str), fast_tokenizer_distilbert, max_length=MAX_LEN)
     4 val_x = fast_encode(validation_set['question_text'].astype(str), fast_tokenizer_distilbert, max_length=MAX_LEN)
     5 train_y = train_set['target'].values
     6 val_y = validation_set['target'].values
     7 print(train_x.shape)
     8 print(train_y.shape)
     9 print(val_x.shape)
    10 print(val_y.shape)
```

Figure 6: Tokenizing the sentences for BERT based Models

(e) *Padding Sentences*: Since the deep learning algorithms consider the data to be represented in vectors, the tokens of each question are to be sequenced into the same length. This allows the deep learning algorithms to efficiently perform matrix operations and provide better results. The padding of the tokens is performed using the pad_sequences function of the keras library.

(f) *Converting Data into tensorflow compatible format*: Since the transformers based models utilizes tensorflow framework, The data needs to be converted into compatible format. The same has been carried out using code shown in Figure 7.

```
[ ]  1 #Converting datasets in order to make it compataible with Tensorflow
     2
     3 train_dataset = (
     4     tf.data.Dataset
     5     .from_tensor_slices((train_x, train_y))
     6     .repeat()
     7     .shuffle(2048)
     8     .batch(BATCH_SIZE)
     9     .prefetch(AUTO)
    10 )
    11
    12 valid_dataset = (
    13     tf.data.Dataset
    14     .from_tensor_slices((val_x, val_y))
    15     .batch(BATCH_SIZE)
    16     .cache()
    17     .prefetch(AUTO)
    18 )
    19 print(train_dataset)
    20 print(valid_dataset)
```

Figure 7: Converting data into tensorflow compatible format

3.4 Modelling

This stage of the CRISP-DM methodology emphasizes the selection of specific modelling techniques to be adopted in the project to attain the desired result. It has been proved by author's of (8) that Recurrent Neural Network based Neural Network models are more efficient in coping with NLP applications than their Convolutional Neural Network based counterparts which in themselves outperform the conventional machine learning algorithms. The study conducted by author's of (27) accentuated the fact that Bi-directional layers outperform normal feed-forward layers in RNN based Neural Network models. Also, the research done by authors of (28) demonstrated that the Binary

Encoder Representation from Transformers (BERT) based models have performed exceptionally well in the sarcasm detection application. Based on the findings of these studies, two different categories of models were finalized to be applied in this research:

1. Recurrent Neural Network based Deep Learning Models:
   (a) Model-1 (Bi-directional GRU Layer)
   (b) Model-2 (Bidirectional LSTM Layer)
   (c) Model-3 (Hybrid model containing Bidirectional GRU & Bidirectional LSTM layer)
   (d) Model-4 (Bi-directional GRU Layer with Attention Layer)
   (e) Model-5 (Bi-directional LSTM Layer with Attention Layer)
   (f) Model-6 (Hybrid Model containing Bi-directional GRU Layer, Bi-directional LSTM Layer and Attention layer)
2. Binary Encoder Representation form Transformers (BERT) based Models:
   (a) Model-7 (DistilBERT)
   (b) Model-8 (RoBERTa)
   (c) Model-9 (BERT)

3.5 Evaluation

This is the penultimate stage in the CRISP-DM methodology which deals with the section of metrics for the evaluation of the performance of the models. The performance of the models considered in this research project will be evaluated and compared on the metrics of the F1 score which is essentially the harmonic mean of precision and recall statistics generated by the models. The advantage of using these metrics is that due to the presence of harmonic mean in the calculation, the influence of extreme outliers can be nullified and relatively unbiased results could be obtained (29).

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{1}$$

3.6 Deployment

This is the last stage of the cRISP-DM methodology which focuses on the deployment aspects of the project for real-world application and resolving the problem identified at the first stage i.e., business understanding. The complexity of this stage varies depending on the requirement of the business. It could either be a single step of generating a report to repeating the whole data mining process across every department of the enterprise (26).

4 Design Specification

A three-tier framework has been adopted in the classification of questions in online community Q&A forums using NLP with RNN based models and Transformer based models as shown in Figure 8. This design highlights the various stages in the implementation aspect of this research project and consists of three different tiers as follows: • Data Layer • Business Logic Layer • Evaluation Layer The Data layer covers the operations carried out related to the collection of data from the source, cleaning the data to remove missing values, preprocessing and tokenization of data using python libraries such as sklearn, keras, matplotlib etc. through a Graphics Processor Unit (GPU) and Tensor Processing Unit (TPU) enhanced Jupyter notebook in Google Colab for RNN based Deep learning models and Transformers based models respectively. The Business logic layer covers all the RNN and Transformers based deep learning models used for the question classification application. The Evaluation layer encapsulates the evaluation of the performance generated by all the deep learning models

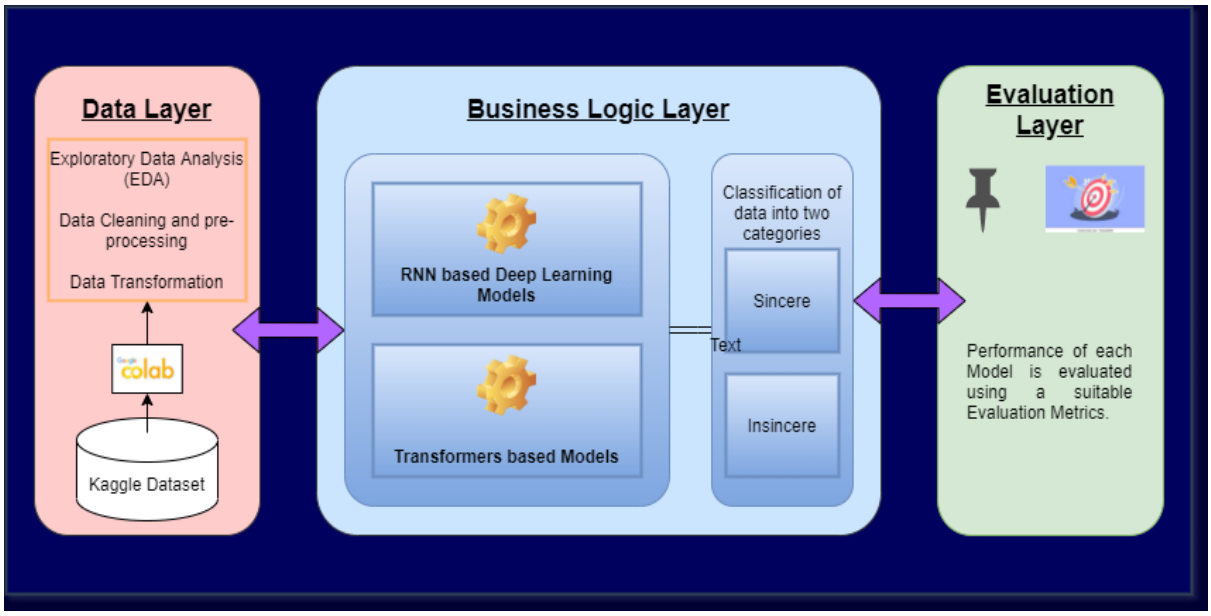


Figure 8: 3-tier Design Framework adopted for this research project

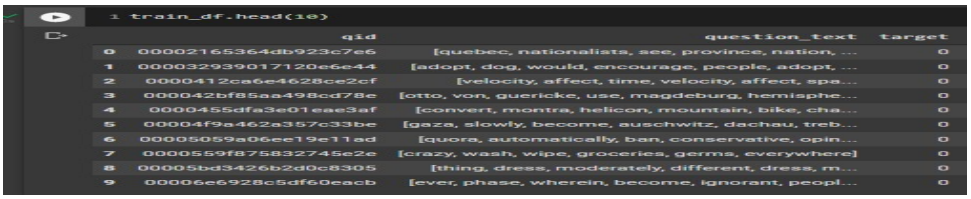
5 Implementation

In this section of the report, the implementation aspect of all the RNN based Deep Learning models, as well as transformers-based models in the classification of questions application of NLP, has been discussed in detail. It also covers the initial exploration of data alongside the pre-processing steps carried out to get the data ready to be fed into the models.

5.1 Data Pre-processing

The most crucial objective and one of the most fundamental aspects of the research is to process the data into an adequate version suitable for the models to train and provide

meaningful results. The dataset consists of various questions asked by users on Quora, an online community Q&A forum. However, the data needs to be cleaned and prepared for the models to provide significantly relevant results. Therefore, all the conjunction, preposition and special characters from the question_text attribute of the data and all the contracted words such as they've, they're, aren't etc. have been expanded back to their original forms. Next, all the stopwords have been removed and the words have been lemmatized to retain their root words only. The cleaned and processed dataset has been shown in Figure 9.



```
1 train_df.head(10)
```

	qid	question_text	target
0	00002165364db923c7e6	[quebec, nationalists, see, province, nation, ...	0
1	000032939017120e6e44	[adopt, dog, would, encourage, people, adopt, ...	0
2	0000412ca6e4628ce2cf	[velocity, affect, time, velocity, affect, spa...	0
3	000042bf85aa498cd78e	[otto, von, guericke, use, magdeburg, hemisphe...	0
4	0000455dfa3e01eae3af	[convert, montra, helicon, mountain, bike, cha...	0
5	00004f9a462a357c33be	[gaza, slowly, become, auschwitz, dachau, treb...	0
6	00005059a06ee19e11ad	[quora, automatically, ban, conservative, opin...	0
7	0000559f875832745e2e	[crazy, wash, wipe, groceries, germs, everywhere]	0
8	00005bd3426b2d0c8305	[thing, dress, moderately, different, dress, m...	0
9	00006e6928c5df60eacb	[ever, phase, wherein, become, ignorant, peopl...	0

Figure 9: Data after pre-processing

5.2 Exploratory Data Analysis

Further exploration of the data was carried out to obtain the following conclusions through the visualizations generated: From Figure 10, it was concluded that the dataset is imbalanced and is skewed towards sincere questions which essentially replicate the real-world situation

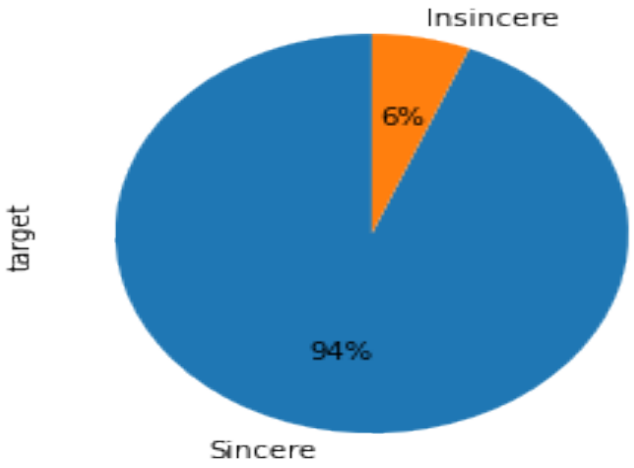
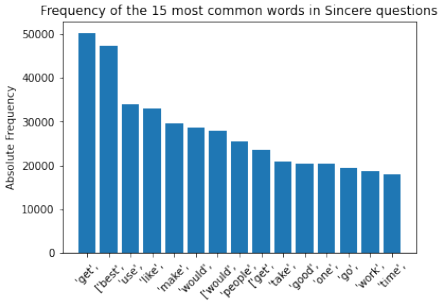


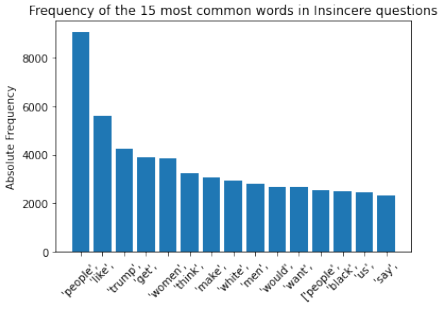
Figure 10: Category distribution of data

Figure 11 and Figure 12 shows the most frequent words that appeared in Sincere and Insincere questions respectively. Similarly, Figure 13 and Figure 14 shows the comparison of the relative frequency of most appeared words in both Sincere and Insincere questions respectively.



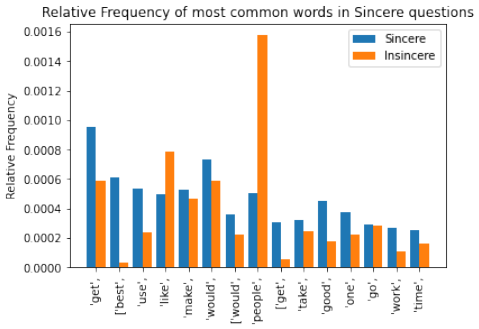
1.0

Figure 11: Common words in Sincere category



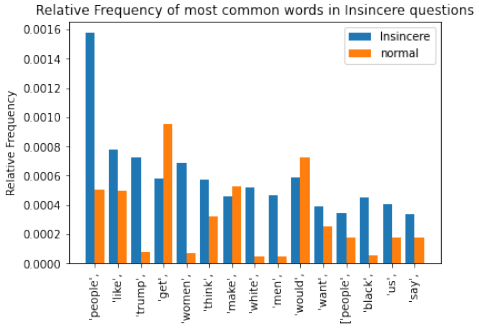
1.0

Figure 12: Common words in Insincere category



.5

Figure 13: Relative frequency check between Common words in Sincere category



.5

Figure 14: Relative frequency check between Common words in Insincere category

Figure 15 and Figure 16 shows the WordCloud presentation of the most frequently appearing words in sincere and insincere questions respectively.

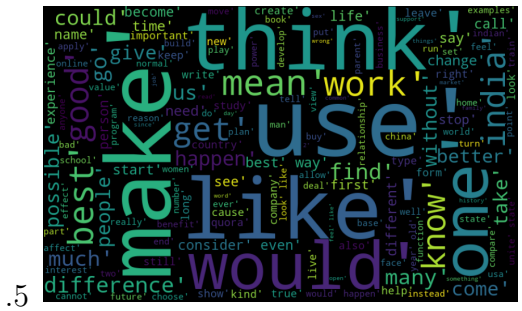


Figure 15: Wordcloud of words in Sincere category

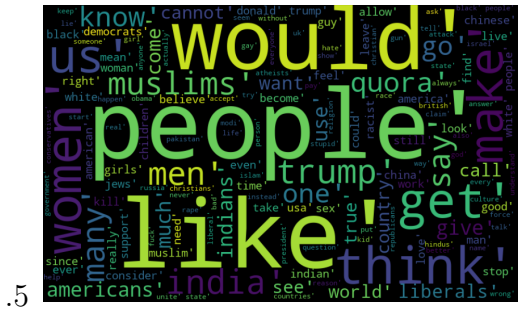


Figure 16: Wordcloud of words in Insincere category

5.3 Implemented Models

Based on the previous studies carried out by [RIC_4], it was observed that the RNN based Deep Learning models have outperformed other architecture based Neural Network models. However, RNN layer architecture was exposed to vanishing and exploding gradient problems. The Long Short-Term Memory layer was specifically designed to overcome this issue. Furthermore, to reduce the computational power required by the LSTM layer-based RNN models, Gated Recurrent Unit (GRU) layer was introduced. It was also observed in previous studies that the inclusion of the Attention layer within RNN architecture-based Deep Learning models marginally improve their performance. Also, Binary Encoder Representation from Transformers (BERT) based models have shown decent promise in the field of NLP application recently.

5.3.1 Recurrent Neural Network based Deep Learning Models

1. *Model-1 (Bi-directional GRU layered RNN based Deep Learning Model)* This model consists of 7 layers. It contains an input layer which is immediately followed by an embedding layer. The output of this embedding layer is then fed into the Bi-directional GRU layer which is then followed by the max_pooling layer. These layers extract the relevant linguistic features and feed them to the dense layer which is then followed by a dropout layer. A final dense layer feeds on the output of the dropout layer and provides the final classification result. Initially, this model was trained for 5 epochs over multiple sets of batch_size. It essentially utilizes the keras library of Python in Google Colab. During Hyperparameter tuning, it was observed that the best performance of the model peaked in the 2nd epoch at batch_size of 256 and started to decrease thereafter. Hence, the final version of the model works over a batch_size of 256 and up to 2nd epoch only.

2. *Model-2 (Bi-directional LSTM layered RNN based Deep Learning Model)* This model has a very similar architecture to that of Model-1. The only difference being that this model utilizes a Bi-directional LSTM layer in place of the Bi-directional GRU layer in Model-1. The function of each layer is the same as that in Model-1. This model was initially trained for 7 epochs over various ranges of batch_size. This model utilizes the keras layer of Python in Google Colab. The best results of this model were achieved with batch_size of 256 and on the 2nd epoch.

3. *Model-3 (Bi-directional GRU + Bi-directional LSTM layered RNN based Deep Learning Model)* This model consists of 14 layers. The initial layers of the models are similar to that of Model-1 and Model-2, however, it utilizes both GRU and LSTM layers in respective order followed by an average pooling layer and max-pooling layer whose outputs are then concatenated before feeding them to two layers of dropout and dense layers. This particular model was initially trained for 10 epochs with a range of batch_size. The best results were obtained with a batch_size of 256 and on the 2nd epoch.

4. *Model-4 (Bi-directional GRU + Attention layered RNN based Deep Learning Model)* It was observed in previous studies (13) that the conjunction of the Attention layer with an RNN architecture model, slightly improves the performance of the model. This particular model consists of 8 layers. It consists of an Attention layer sandwiched between the Bi-directional GRU layer and the first dropout layer which boosts up the performance in identifying linguistic relations between tokens. This model was also trained over a range of batch_size with 5 epochs. The final model was selected at batch_size of 256 having 2 epochs as the performance of the model was found to be at its best with these values of parameters.

5. *Model-5 (Bi-directional LSTM + Attention layered RNN based Deep Learning Model)* The architecture of this model is very similar to that of Model-4 with the exception that the Bi-directional LSTM layer replaces its Bi-directional GRU layer. Also, the function of layers remains essentially the same as that of Model-4. This model was also trained over a range of batch_size with 5 epochs. The best performance of the model was observed at batch_size of 256 and on the 2nd epoch.

6. *Model-6 (Bi-directional GRU +Bi-directional LSTM + Attention layered RNN based Deep Learning Model)* The architecture of this model is essentially the same as that of Model-3 with the exception that two Attention layers have been introduced in between the LSTM layer and pooling layer which feeds on the output of the Bi-directional GRU layer and Bi-directional LSTM layer respectively. The outputs of these Attention layers are then fed to pooling layers whose outputs are then further concatenated. This model was trained for a range of batch_size with 7 epochs. The best results were produced by this model with batch_sizeof 256 and on the 2nd epoch.

5.3.2 Binary Encoder Representations from Transformers (BERT) based models

1. *Model-7 (DistilBERTl)* DistilBERT or Distilled version of BERT, is a BERT based model which encapsulates only 60% of the parameters of the BERT model making it almost 60% faster. It also manages to maintain 95% of the performance to that of

BERT and therefore is a great model to be utilized. Hyperparameter tuning of this model resulted in the best performance of the model with batch_size equivalent to 16 times the no. of available tensor processing clusters which are operating in sync and with 3 epochs. This allowed faster batch sampling since it's equally distributed between the tensor processing unit clusters. Also, after 3rd epoch, the performance of the model started to depreciate.

2. *Model-8 (RoBERTa)* Robustly optimized BERT pretraining approach (RoBERTa) is also a BERT based model which advances over its language masking strategy and is developed by Facebook's AI. RoBERTa improves over BERT by training over larger batches and learning rates. The 'next sentence pretraining objective' of BERT was also removed to make RoBERTa faster and require less computational time. Hyperparameter tuning of the RoBERTa model resulted in the same results as that of the DistilBERT model.

3. *Model-9 (BERTl)* Robustly optimized BERT pretraining approach (RoBERTa) is also a BERT based model which advances over its language masking strategy and is developed by Facebook's AI. RoBERTa improves over BERT by training over larger batches and learning rates. The 'next sentence pretraining objective' of BERT was also removed to make RoBERTa faster and require less computational time. Hyperparameter tuning of the RoBERTa model resulted in the same results as that of the DistilBERT model.

6 Evaluation

The evaluation metrics considered for evaluating the results generated by the applied models in this research project is F1-score. This particular metrics has been selected since it nullifies the effect of extreme values due to the presence of harmonic mean of precision and recall metrics in calculation which makes it a perfect to be considered for imbalanced and skewed dataset. The mathematical equation for calculation of F1-score is shown in equation (1).

6.1 Results of RNN based Deep Learning Models

It was observed that the Model-3 (Bi-directional GRU + Bi-directional LSTM layer) outperformed both Model-1 (Bi-directional GRU layer) and Model-2 (Bi-directional LSTM layer), achieving an F1-score of 63.27% closely followed by Model-2 and Model-1 with F1-scores of 62.11% and 62.94% respectively. However, with the introduction of Attention layer, Model-6 (Bi-directional GRU + Bi-directional LSTM + Attention layer) performed all the RNN based Deep learning models implemented and achieved an F1-score of 63.53% closely followed by Model-4 (Bi-directional GRU + Attention layer) and Model-5 (Bi-directional LSTM + Attention layer) with F1 score of 63.36% and 63.53% respectively.

The results achieved by all the RNN based Deep Learning models is presented in Table 1.

Table 1: Results of all the RNN based Deep Learning Models

Model	Description	F1-score achieved
Model-1	Bi-directional GRU layer	62.94%
Model-2	Bi-directional LSTM layer	63.11%
Model-3	Bi-directional GRU + Bi-directional LSTM layer	63.27%
Model-4	Bi-directional GRU + Attention layer	63.36%
Model-5	Bi-directional LSTM + Attention layer	62.74%
Model-6	Bi-directional GRU + Bi-directional LSTM + Attention layer	**63.53%**

6.2 Results of Transformers based Models

It was observed that Model-7 (DistilBERT) outperformed both Model-8 (RoBERTa) and Model-9 (BERT) and achieved an F1 score of 94.87% followed by Model-9 (BERT) and Model-8 (RoBERTa) with F1-scoresof 94.21% and 75.63% respectively. The results achieved by all the Transformers based models is presented in Table 1.

Table 2: Results of all the Transformers based Models

Model	Description	F1-score achieved
Model-7	DistilBERT	**94.87%**%
Model-8	RoBERTa	75.63%
Model-9	BERT	94.21%

6.3 Discussion

Similar classification problem has been addressed by (12) and their study achieved highest F1 score of 87.81% using Multi-Layered Perceptron model with POSTAG feature. This research study proves that the results of this classification application of NLP can be improved using Transformers based Models. Two of the three such models i.e., Model-7 (DistilBERT) and Model-9 (BERT) implemented in this study have outperformed the results achieved by previous studies with a margin of 7.06% and 6.4% respectively.

Table 3 shows the Comparison of performance obtained by all the models in this research project with previous studies.

Table 3: Comparing performance of all models implemented in this study with previous studies.

Model	**Description**	**F1-score achieved**
Previous study	MLP with POSTAG Feature	87.81%
Model-1	Bi-directional GRU layer	62.94%
Model-2	Bi-directional LSTM layer	63.11%
Model-3	Bi-directional GRU + Bi-directional LSTM layer	63.27%
Model-4	Bi-directional GRU + Attention layer	63.36%
Model-5	Bi-directional LSTM + Attention layer	62.74%
Model-6	Bi-directional GRU + Bi-directional LSTM + Attention layer	**63.53%**
Model-7	DistilBERT	**94.87%**%
Model-8	RoBERTa	75.63%
Model-9	BERT	94.21%

7 Conclusion and Future Work

This research study concentrated on identifying the extent to which question classification application of NLP can be improved using the Transformers based approach when compared to RNN based Deep learning models. RNN based deep learning algorithms have proved themselves to be the best in providing consistent and reliable results in multiple studies. The study observed that the performance of the model having a Bi-directional GRU layer is slightly lower when compared to that of the model consisting of a Bi-directional LSTM layer in classification applications. However, a model containing a combination of these two layers outperformed the results obtained by models which contain them separately by a very small margin. It was also noticeable that with the integration of the Attention layer, the performance of the model with Bi-directional GRU Layer marginally improved in classification application while an opposite trend was observed in the performance of model containing Bidirectional LSTM layer. Moreover, the model containing both the layers in integration with the Attention layer outperformed all the RNN based Deep Learning models implemented in this research. The models adopting a transformer-based approach in question classification application of NLP heavily outperformed the RNN based Deep learning models and achieved excellent results. It was worth noting that the DistilBERT model outperformed the BERT model and the RoBERTa model. The BERT model required the most computational power and took the longest time in training followed by the RoBERTa model. The DistilBERT model was the fastest and produced the best results among the three. In future, a custom transformer-based architecture could be developed specifically tailored to excel in this particular application. Also, This study only explored the supervised learning field and utilized labelled data only. In future, unsupervised learning using unlabelled data can be explored.

)

References

[1] Y. Lan, Y. Hao, K. Xia, B. Qian, and C. Li, "Stacked residual recurrent neural networks with cross-layer attention for text classification," *IEEE Access*, vol. 8, pp. 70 401–70 410, 2020.

[2] R. G. Athreya, S. K. Bansal, A. C. N. Ngomo, and R. Usbeck, "Template-based question answering using recursive neural networks," in *2021 IEEE 15th International Conference on Semantic Computing (ICSC)*, 2021, pp. 195–198.

[3] Y. Cao, M. Li, T. Feng, R. Wang, and Y. Wu, "Improving question classification with hybrid networks," in *2019 International Conference on Asian Language Processing (IALP)*, 2019, pp. 166–171.

[4] Z. Yang, L. Wang, and Y. Wang, "Multi-intent text classification using dual channel convolutional neural network," in *2019 34rd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, 2019, pp. 397–402.

[5] C. I. Eke, A. A. Norman, and L. Shuib, "Context-based feature technique for sarcasm identification in benchmark datasets using deep learning and bert model," *IEEE Access*, vol. 9, pp. 48 501–48 518, 2021.

[6] D. Kim, J. Koo, and U. M. Kim, "Envbert: Multi-label text classification for imbalanced, noisy environmental news data," in *2021 15th International Conference on Ubiquitous Information Management and Communication (IMCOM)*, 2021, pp. 1–8.

[7] A. Agarwal, V. Kumari, Y. Sharma, and L. Goel, "Ranking based question answering system with a web and mobile application," in *2021 11th International Conference on Cloud Computing, Data Science Engineering (Confluence)*, 2021, pp. 52–58.

[8] A. Dhakal, A. Poudel, S. Pandey, S. Gaire, and H. P. Baral, "Exploring deep learning in semantic question matching," in *2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS)*, 2018, pp. 86–91.

[9] R. Anhar, T. B. Adji, and N. Akhmad Setiawan, "Question classification on question-answer system using bidirectional-lstm," in *2019 5th International Conference on Science and Technology (ICST)*, vol. 1, 2019, pp. 1–5.

[10] D. A. Prabowo and G. Budi Herwanto, "Duplicate question detection in question answer website using convolutional neural network," in *2019 5th International Conference on Science and Technology (ICST)*, vol. 1, 2019, pp. 1–6.

[11] J. X. Chen, D. M. Jiang, and Y. N. Zhang, "A hierarchical bidirectional gru model with attention for eeg-based emotion classification," *IEEE Access*, vol. 7, pp. 118 530–118 540, 2019.

[12] H. M. Lynn, S. B. Pan, and P. Kim, "A deep bidirectional gru network model for biometric electrocardiogram classification based on recurrent neural networks," *IEEE Access*, vol. 7, pp. 145 395–145 405, 2019.

[13] N. Colnerič and J. Demšar, "Emotion recognition on twitter: Comparative study and training a unison model," *IEEE Transactions on Affective Computing*, vol. 11, no. 3, 2020.

[14] D. Chen, X. Yan, X. Liu, S. Li, L. Wang, and X. Tian, "A multiscale-grid-based stacked bidirectional gru neural network model for predicting traffic speeds of urban expressways," *IEEE Access*, vol. 9, pp. 1321–1337, 2021.

[15] Q. Tao, F. Liu, Y. Li, and D. Sidorov, "Air pollution forecasting using a deep learning model based on 1d convnets and bidirectional gru," *IEEE Access*, vol. 7, pp. 76 690–76 698, 2019.

[16] B. Pang and L. Lee, *A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts.In Proceedings of the 42nd annual meeting on association for computational linguistics, Barcelona, Spain (p.271)*, 2004, cited By :1. [Online]. Available: www.scopus.com

[17] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *EMNLP 2013 - 2013 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 2013, pp. 1631–1642, cited By :2427. [Online]. Available: www.scopus.com

[18] P. Semberecki and H. Maciejewski, "Deep learning methods for subject text classification of articles," in *2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2017, pp. 357–360.

[19] A. F. Adoma, N.-M. Henry, and W. Chen, "Comparative analyses of bert, roberta, distilbert, and xlnet for text-based emotion recognition," in *2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, 2020, pp. 117–121.

[20] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2019.

[21] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," 2020.

[22] M. Naseer, M. Asvial, and R. F. Sari, "An empirical comparison of bert, roberta, and electra for fact verification," in *2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, 2021, pp. 241–246.

[23] J. Thorne, A. Vlachos, O. Cocarascu, C. Christodoulopoulos, and A. Mittal, "The fact extraction and verification (fever) shared task," 2018.

[24] M. U. Sarwar, S. Zafar, M. W. Mkaouer, G. S. Walia, and M. Z. Malik, "Multi-label classification of commit messages using transfer learning," in *2020 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, 2020, pp. 37–42.

[25] Y. Chen and Q. Song, "News text summarization method based on bart-textrank model," in *2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, vol. 5, 2021, pp. 2005–2010.

[26]

[27] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, *Recursive Deep Models for Semantic Compositionality over A Sentiment Treebank*, pp. 1–6, 2015, cited By :1. [Online]. Available: www.scopus.com

[28] T. Orth and M. Bloodgood, "Early forecasting of text classification accuracy and f-measure with active learning," in *2020 IEEE 14th International Conference on Semantic Computing (ICSC)*, 2020, pp. 77–84.

[29] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, "Advances in pre-training distributed word representations," in *LREC 2018 - 11th International Conference on Language Resources and Evaluation*, 2019, pp. 52–55, cited By :210. [Online]. Available: www.scopus.com