EasyChair Preprint

№ 7283

Tweet Sentiment Extraction

Chiranjeev Mishra, Eshit Bansal and A. Helen Victoria

January 5, 2022

Tweet Sentiment Extraction

Chiranjeev Mishra
Dept. of Networking & Communication
SRM Institute of Science & Technology
Kattankulathur, Tamil Nadu, 603203
cm4004@srmist.edu.in

Eshit Bansal
Dept. of Networking & Communication
SRM Institute of Science & Technology
Kattankulathur, Tamil Nadu, 603203
eb5153@srmist.edu.in

Mrs. A. Helen Victoria
Dept. of Networking & Communication
SRM Institute of Science & Technology
Kattankulathur, Tamil Nadu, 603203
helenvia@srmist.edu.in

***Abstract*—The classic sentiment analysis problem deals with analysing the overall polarity of a set of responses. Though by only knowing the polarity, an organisation can't get an idea about why they received such responses. Thus this makes them unable to analyse the responses, which could have possibly helped them in the betterment of the service they were providing.**

The purpose of this project was to make a question answering model which would extract a phrase out of a given tweet which amplifies a given sentiment (positive/negative/neutral). Using initial training and testing runs which were scored using Jaccard score, we compared the performance between BERT (standard method), RoBERTa, DistilBERT and AlBERT to find out the best performing method on the given Twitter dataset. After, DistilBERT was found out to give the best performance out of the above mentioned methods with an accuracy of 68.92% over BERT's accuracy of 64.57%, it was then further fine-tuned by pre-processing, processing and post-processing methods to make a final model which gave an accuracy of 73.12%.

The project successfully implements a model which can extract phrases out of a given text (in this case tweets), the current accuracy benchmark for which is 73.12%. Further optimisation is required to increase the accuracy even more, so that it can replicate BERT's performance of 85% accuracy which it achieved on the SQUAD dataset.

***Keywords*—Sentiment Analysis, NLP, extract a phrase, Jaccard score, BERT, DistilBERT, question answering.**

I. INTRODUCTION

A. Prologue

Sentiment Analysis refers to the identifications, classifications, and extractions of emotions in given text using NLP techniques.

The classical problem involves with calculating the polarity of a given text – it can be something ranging from extremely negative (the 0 mark in numerical terms) to extremely positive (the 1 mark in numerical terms) with neutral being in between of the two extremes (the 0.5 mark in numerical terms). Here, more the polarity value is towards the 0 mark, more it can be termed as a negative text, and more the polarity value is towards the 1 mark, more it can be termed as a positive text. At the same time crisp values (negative, neutral, positive) are also usually assigned to the text so as to give an overview of the text and not a detailed analysis (in case that is what the user wants to know).

B. Types of Sentiment Analysis

The following are a few types of sentiment analysis –

- **Feature/Aspect based** – It refers to determining the opinions or sentiments expressed on different features of the entity in question.
  It focuses on mining features or aspects of entities (e.g. products) or topics on which people have expressed their opinions and determine whether the opinions are positive or negative.
  For example, instead of analysing the complete set of reviews about a particular movie, the task would first categorise all the different reviews on the basis of the genre about which the reviewer has mainly talked about and then specifically analyse the positivity/negativity of the reviews of that particular genre. Thus giving a better idea and analysis about every aspect (genre) of the movie [1].
- **Multilingual sentiment analysis** – The majority of research efforts for sentiment analysis just like other NLP tasks has been done for the English language, while a great share of information is available in other languages as well. During the years 2009–2015, the number of publications on English sentiment analysis has been 10–40 times more in number to that of the publications about multilingual sentiment analysis [2]. Thus one of the biggest challenges for sentiment analysis is that it is highly language dependent.
  Word embeddings, sentiment lexicons etc are language specific and further, optimizing the models for each language is very time consuming and labour intensive. Thus developing a language independent model which can be reused for different languages irrespective of the type or nature of the language is nowadays a popular problem and is being readily worked upon [3].
- **Emotion recognition** – Human emotion can be classified into a number of different emotions like anger, sadness, joy, love etc. Thus sentiment analysis can also be done on the basis of the different emotions so as to extract emotion from a text. Emotions can be extracted from two essential text forms: written text and conversations (dialogues). For written texts, models usually aim to focus the "word/phrases" representing the emotions [4].

C. Motivation

The classical problem related to sentiment analysis deals with categorising the given text (document, reviews, conver-

sation etc) into neutral, negative or positive sentiments. The classification though does not give any idea about the exact reason for a particular categorisation.

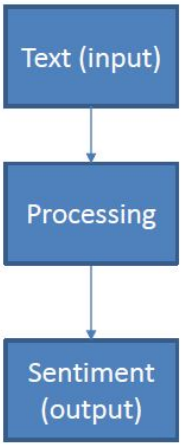

Fig. 1. Flow chart displaying the input and output parameters required for sentiment analsysis

When analysing a set of feedbacks for a product, the overall sentiment based analysis can give an idea about the polarity of all the feedbacks. This polarity score though by itself is unable to give the organisation an idea about why exactly did they receive such a score. For analysing the problems of the products they would thus have to analyse the phrases of the feedbacks, which would then in turn help them to devise potential strategies to solve those problems. For organisations which receive tens of thousands of responses (comments, reviews etc) everyday, a manual analysis is very costly both in terms of time and money. If the organisation is not able to address these issues in an efficient and timely manner then there is a chance that the customer may move on with a product of a different company which would essentially over time result in the loss of the company.

For companies which provide sales, product development, after sales services; it is very important to know the feedback of the customers. The companies use the data collected from their social media accounts (comments received under a particular post), to check why exactly did they receive a positive or a negative reaction. The analyser can likewise be used for analysing government policies, given that in today's time everyone wants to give a comment on why they think that a particular policy is good or bad. It can be used inside an organisation to analyse the reviews about the organisational workflow, the reviews would thus help the leaders of the organisation to improve the quality of the workflow of their organisation, thus improving the work culture.

II. LITERATURE SURVEY

A. Question Answering

For extracting a phrase out of a text which amplifies a given sentiment we need to pose a question to the computer where we would ask in natural language that "What is negative in this

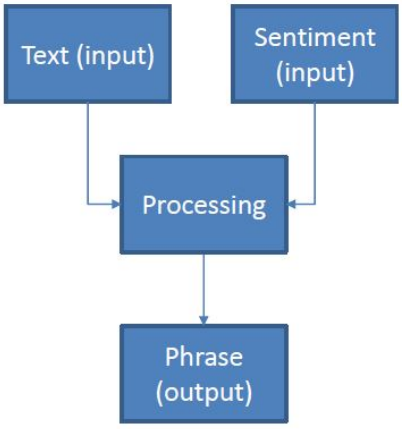

Fig. 2. Flow chart displaying the input and output parameters required for sentiment phrase extraction

text?". The computer would then try to answer by checking the text, and find the relevant words which can possibly answer the question posed against it.

Such a question can be answered by a model which is made using QA. QA is a computer science sub-discipline which lies within the fields of NLP. It focuses on building systems that can answer the questions asked by a human in natural language [5].

QA research attempts to deal with a variety of question types including: definition, fact, How, Why, list, cross lingual questions etc. For example if a question is asked to a QA model –

What is the capital of India?

then the model would answer '*New Delhi*'.

B. Question Answering Methods

There are mainly two types of question answering methods, the first one being open domain question answering and the second one being mathematical question answering –

- **Open domain question answering** – An open domain question answering system aims at returning an answer in response to a user's question. The returned answer is in the form of short texts rather than a list of relevant documents [6].
  The system uses a combination of knowledge representation, computational linguistics and information retrieval to find an answer.
- **Mathematical question answering** – An open math-aware question answering system returns a single mathematical formula for a natural language question. These formulae originate from the knowledge-based Wikidata. The authors translate these formulae to compatible data by integrating the calculation engine sympy into the system. This way, users can enter numeric values for variables occurring in the formula [7].

TABLE I
TABLE DEPICTING THE INPUT TRAINING DATASET

S. No.	textId	text	Sentiment	selected_text
1.	cb774db0d1	I'd have responded, if I were going	neutral	I'd have responded, if I were going
2.	549e992a42	Sooo SAD I will miss you here in San Diego!!!	negative	Sooo SAD
3.	6e0c6d75b1	2 am feedings for the baby are fun when he is all smiles and coos	positive	Fun
4.	e050245fbd	Both of you	neutral	Both of you
5.	04dd1d2e34	i want to go to music tonight but I lost my voice.	negative	lost

C. Previous attempts on sentiment phrase extraction

In the last few years, a lot of work has been done for both regular question answering and regular sentiment analysis, but very little work has been done on sentiment phrase extraction. Following are some reports which have been made for sentiment phrase extraction –

- **Agarwal et al.** [8] extracted phrases using a POS-based fixed indicator. The syntactic relation created from the text was then used to create a dependency tree to get the appropriate phrases for a given sentiment.
  According to the authors, the syntactic patterns are very effective for subjective detection. One major limitation of using this approach as mentioned in the paper by the authors is that the phrases extracted using the POS-based fixed patterns are not efficient in extracting sentiment-rich phrases. The phrases extracted are not entire phrases, and are instead only a group of one or two words (which the resultant phrase may not be in all the cases).
- **Vu et al.** [9] propose an approach which deals with making a phrase template which is a sequence of POS tags corresponding to a large number of valid phrases in English. The extracted phrase templates are then clustered into more coherent topics for tracking. The topics here can be the different domains on which a given set of text (comments or replies) can be grouped upon.
  According to the authors, when applying the approach to two case studies of real life problems, the approach helped in detecting major user opinions. Though at the same time they also mentioned how the case study only had limited data and thus the validation of if the approach is scalable or not is something which is required to be researched upon. Another problem this approach faced was that the approach was language-dependent, so if it works for one language (usually english), for the approach to work on other languages the respective POS information also has to be found (which is usually not very easy to find).

The two state-of-the-art systems based on QA model are Google's BERT and IBM's Watson. These models have been benchmarked on various popular datasets but still the results/performance of these state-of-the-art systems cannot be generalised such as our tweet sentiment extraction dataset. Hence, there is a need to customise the state-of-the-art systems to match our specific requirement of sentiment extraction from tweets.

III. METHODOLOGY AND RESULTS

This section presents the appraoch proposed to make the model. The whole process was divided into multiple steps -

A. Data Acquisition

The dataset was downloaded from Figure Eight's data for everyone platform. The dataset was titled 'Sentiment Analysis: Emotion in Text tweets with existing sentiment labels'. It contained a set of 27,480 data points which contain a 'textId', a given 'text' and a 'sentiment' along with the respective 'selected_text' which maximised the given sentiment in the given text. An analysis was done on the given dataset so as to decide the appropriate split for the training and testing phase. Table I depicts the input training dataset.

B. Selection of the initial model

This step involves with finding the model which gives the best accuracy when subjected to the training and testing dataset mentioned in the previous step. The four models used are –

- Google's BERT
- Hugging Face's DistilBERT
- Google's AlBERT
- Facebook's RoBERTa

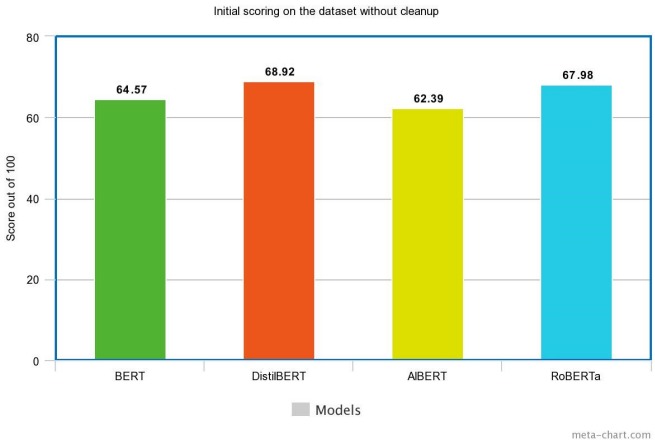

Fig. 3. Graph displaying the accuracy comparison between the four initial models

The training data was first converted in the respective formats which the models allow. For example for DistilBERT, the format involves with converting the list data imported from the CSV file to a dictionary format (key-value pairs). The dictionary would contain the keys 'context' which stores the

TABLE II
TABLE DEPICTING THE TESTING DATASET

S. No.	textId	text	sentiment
1.	8a939bfb59	Uh oh, I am sunburned	negative
2.	4f5267ad70	That's it, it's the end. Tears for Fears vs Eric Prydz, DJ Hero http://bit.ly/2Hpbg4	neutral
3.	2724775d6b	Born and raised in NYC and living in Texas for the past 10 years! I still miss NY	negative
4.	95e12b1cb1	He's awesome...Have you worked with him before? He's a good friend	positive

values of the column 'text' and 'qas' which further contains keys which store the value of the 'textId', 'question' and 'answer'. The question here being "What is negative in this tweet?" or "What is positive in this tweet?. The 'answer' contains the value from the corresponding 'selected_text' column.

After the data was converted into the required format, the hyper-parameters were required to be set. Hyper-parameters like the number of epochs, the learning rate, doc-stride were changed manually by trying different value pairs. In the end, the best observed accuracy for each hyper-parameter pair for each model was recorded. The accuracy of the four models is compared in Figure. 3.

C. Testing using Jaccard Score

Jaccard score was used as the metric to calculate the accuracy of the output presented by the models in the previous step, and to also calculate the accuracy of the model being fine-tuned over the course of the project.

It is calculated by using the formula

$$\frac{\textit{No.of elements in the intersection}}{\textit{No.of elements in the union}} \tag{1}$$

For example assuming we have two statements, first being the 'selected_text' or the ground truth of the particular 'text'-'sentiment' pair, and the second being the prediction made by the model for the given 'text'-'sentiment' pair. So if the 'selected_text' is

Hello, my name is Shubham

and the predicted text is –

Hello, my name is Kamboj

then the intersections between the two statements are 'Hello,', 'my', 'name' and 'is', and the elements after the union of two sentiments are 'Hello,', 'my', 'name', 'is', 'Shubham' and 'Kamboj'. The number of elements in the intersection is 4 and the number of elements in the union is 6. Therefore by the formula for the Jaccard score, the accuracy of the predicted text will be 4/6 = 0.67.

This value is calculated for all the data points in the testing dataset with the final accuracy being calculated by averaging the individual accuracies. Table II depicts a sample of the testing dataset which was used to find the predictions and was then evaluated using the Jaccard score.

D. Fine-tuning of the selected initial model (DistilBERT)

1) Pre-processing: When observing the dataset we observed how most of the data in the 'selected_text' is a bit clean than what is being feeded to it. So for improving the accuracy of the model a pre-processing method was added to it. The pre-processing method would thus clean up the http links, emoji and other inconsistencies in the input data so as to present a much cleaner outlook, before feeding the data for training. The cleaner data would thus in turn help in better model generation which would in turn generate better prediction for a 'text'-'sentiment' pair. For the cleaning part, regular expressions were made to catch the target strings in the input data. The target strings were then replaced by empty strings, thus giving us cleaned input data.

When only applying the pre-processing method, the accuracy of the DistilBERT model increased from 68.92% to 71.68%. Thus this method was used in the final model.

2) Processing: Since a QA model was being developed, it was hypothesized if increasing the number of questions can possibly increase the accuracy of the model. Currently only three types of questions were available – positive, negative and neutral. To get a better classification of the above mentioned sentiments, the data points were further classified into more questions on the basis of the text length. Values below a particular threshold, were classified as [sentimet]1 type of question and values above the given threshold were classified as [sentiment]2 type of questions. The neutral sentiment was not classified in this manner as it was already providing nearly 98% accuracy for the neutral marked data points.

The model was then trained using the updated classification of the sentiments and the output was scored. Though when only using this method, the accuracy of the model decreased from a value of 68.92% to 68.71%. The method was thus not used in the final model.

3) Post-Processing: This step involves observing the input and output dataset and making inferences so as to manipulate the output generated by the model to in turn give an increased accuracy.

- It was observed that in the input dataset, the input 'text' was similar to the 'selected_text' quite frequently. On the other hand, in the output dataset the frequency of such a pattern being was quite low. Thus an analysis on the basis of the sentiments presented in the output data was made, and it was found out that if for a neutral sentiment, the output text is 90% similar to the input text then using the full input may instead improve the score. Similar observation was made for the negative sentiments

where it was observed if the output text is 94% similar to the input text then using the whole text in such a case increases the total accuracy. Thus, when using this method the accuracy of the DistilBERT model increased from 68.92% to 70.43%. The method was thus included in the final model.

- It was observed that in the input dataset, whenever the input text and the selected_text had 4 or less words then the output usually came out to be exactly same as what it was fed. Thus, a post-processing method was introduced, which would recover the original text for all the outputs which had four or less words in the original text. Thus, when using this method the accuracy of the DistilBERT model increased from 68.92% to 69.31%. The method was thus included in the final model.

E. Final Model

The final model was thus created by implementing the pre-processing and both the post-processing methods on the DistilBERT model extracted from the initial model selection. The comparison of the individual accuracies of the different processing methods, along with the final accuracy is showed in Figure 4.

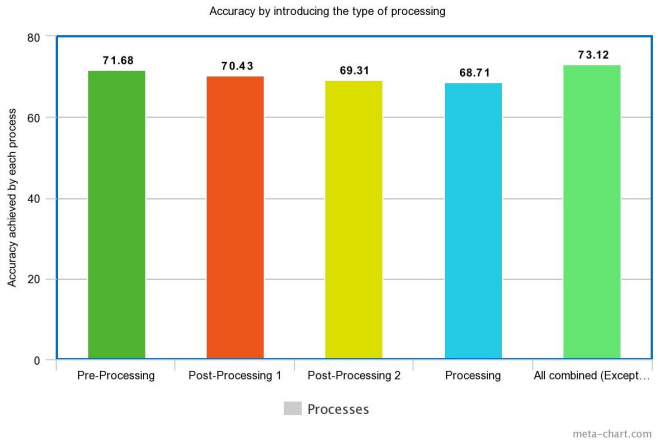

Fig. 4. Graph displaying the accuracy comparison between the different processing methods, along with the final accuracy

F. Block Schematic Diagram

The section briefs the steps used for developing the tweet phrase extractor. Initially, the input dataset was split into two parts – training and testing. The dataset was analysed and an appropriate split value was selected.

The four different question answering methods were then trained on the training dataset and were scored using the Jaccard score, since it punishes the output heavily whenever it underfits or overfits. The performance for the four models was then compared so as to select a model which could be then selectively further fine-tuned for our dataset. During this phase it was observed that DistilBERT gave the best accuracy for the given dataset.

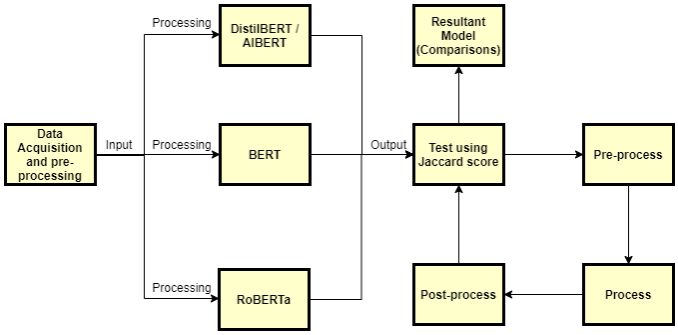

Fig. 5. Block Diagram

DistilBERT was then later subjected to different pre-processing, processing and post-processing methods which were inferred by observing the differences between the input and the output data. After the implementation of the different methods, the model was combined as a whole so as to get a final model. A final accuracy of 73.12% was achieved which was significantly higher than BERT's (state-of-the-art method) accuracy of 64.57%. Thus, the final model has better performance than the state-of-the-art method and would work in a more specific manner for tweets, for phrase extraction.

IV. CONCLUSION

Though BERT has been benchmarked at different infamous datasets, it does not perform well on the Tweets Dataset. Thus selection of an initial model on the basis of the accuracy helped in a better initial model selection which when subjected to different processing methods eventually increased the accuracy of the model initially present on hand. Starting from BERT's accuracy of 64.57%, an accuracy of 73.12% was finally achieved for the given dataset.

Though the tweet sentiment phrase extractor was implemented, it still does not give a very high level of accuracy for the given problem statement. Dividing the given dataset on the basis of length to devise more questions did not work, since creating new questions essentially meant that data for each question type is reducing, which is not good for training purposes. This approach may give a better performance if more data points are provided so that the model may not suffer from data scarcity.

Since the pre and post-processing methods try to generalise the input and output, they at times create an output which is different from the ground truth the model had predicted in the first place. This thus reduces the overall accuracy, and thus results in a decreased accuracy than what the model could have potentially achieved.

An ensemble model can be made in future. Since RoBERTa has a near similar accuracy to that of DistilBERT, it can possibly be used along with DistilBERT to create a better overall model. Depending on the type of sentiment, the model which gives the better accuracy during the testing stage for that particular sentiment can then be used to create the output of that sentiment. The final model can then be a combination

of these different models. Though implementation of such a model would result in more time and memory resources.

REFERENCES

[1] M. Hu and B. Liu, "Opinion mining, sentiment analysis, and opinion spam detection," 2004.

[2] K. Dashtipour, S. Poria, A. Hussain, E. Cambria, A. Y. Hawalah, A. Gelbukh, and Q. Zhou, "Multilingual sentiment analysis: State of the art and independent comparison of techniques," *Cognitive Computation*, p. 757–771, 2016.

[3] E. F. Can, A. Ezen-Can, and F. Can, "Multilingual sentiment analysis: An rnn-based framework for limited data," in *ACM SIGIR Conference*, 2018.

[4] S. N. Shivhare and S. Khethawat, "Emotion detection from text," in *Data Mining and Knowledge Management Process*, vol. 2, 2012. [Online]. Available: https://doi.org/10.5121/csit.2012.2237

[5] P. Cimiano, C. Unger, and J. McCrae, *Ontology-Based Interpretation of Natural Language*, 2014.

[6] H. Sun, B. Dhingra, M. Zaheer, K. Mazaitis, R. Salakhutdinov, and W. W. Cohen, "Open domain question answering using early fusion of knowledge bases and text," in *EMNLP*, 2018. [Online]. Available: https://doi.org/10.18653/v1/D18-1455

[7] M. Schubotz, P. Scharpf, K. Dudhat, Y. Nagar, F. Hamborg, and B. Gipp, "Introducing mathqa - a math-aware question answering system," 11 2018. [Online]. Available: https://doi.org/10.1108/IDD-06-2018-0022

[8] B. Agarwal, V. K. Sharma, and N. Mittal, "Sentiment classification of review documents using phrase patterns," in *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2013, pp. 1577–1580.

[9] P. M. Vu, H. V. Pham, T. T. Nguyen, and T. T. Nguyen, "Phrase-based extraction of user opinions in mobile app reviews," in *2016 31st IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 2016, pp. 726–731.