COVID 19 NLP Text Classification

Nikhil Swami *CSE*
School Of Engineering
nikhil46_soe@jnu.ac.in

Ankit Tiwari *CSE*
School Of Engineering
ankit99_soe@jnu.ac.in

Ayush Bharti *CSE*
School Of Engineering
ayush38_soe@jnu.ac.in

Sayantan Roy *CSE*
School Of Engineering
sayant29_soe@jnu.ac.in

Ajay Verma CSE
School Of Engineering
ajay53_soe@nu.ac.in

***Abstract*—There is so much to a person than just their face. In this paper, we discuss how we've tried different methods to check the sentiments of any person's tweet or sometimes of the entire population by Deep Learning methods. We've mentioned which are the models that we've used, why we've used them and how changing each of the features in a whole affects the overall analysis of the model. This project was mainly carried out in Google's Colab environment, along with Amazon's S3. Finally, we mention the accuracies and attributes of each.**

I. Introduction

The original dataset was provided from Kaggle[1] which contained testing and training datas in comma separated values(csv). Training & testing set contained 41158 and 3799 records. Each of the examples contains 4 columns containing - Location, Tweet device used, Original Tweet and Label associated with it.

Natural Language Processing has significantly evolved during the years. A brief overview of the history behind NLP, arriving at today's state-of-the-art algorithm BERT, and demonstrating how changes have been made into them has been shown in our project where we've used even further evolved models like RoBERTa (from facebook), DistilRoBERTa, as well XLNETs. Our results from the model gives us more than 79% accuracy in total average.

A. Abbreviations and Acronyms

ML (Machine Learning) , AI (Artificial Intelligence) , BERT (Bidirectional Encoder Representations from Transformers), CNN (Convolution Neural Network), LSTM (Long Short Term Memory).

II. Problem Statement

For policy makers as well as for researchers, it is important to know if the general sentiment of the population is positive or not. Knowing overall sentiment will give us a fair idea about how the population reacts to each kind of news article, decisions or any major events like the Coronavirus. We Have done an Exact Implementation Of The Problem Statement report which was asked previously. 4 Models 4 ideas.

To know the impact thus, we've tried making a model that can perform sentiment analysis on each tweet - which can further be developed to give entire mass sentiment. We need to make a model which can predict the sentiment of the tweet. And classify it into Extreme Negative, Negative , Neutral, Positive and Extreme Positive. Perform Text Classification on the data. The tweets have been pulled from Twitter and manual tagging has been done then. The names and usernames have been given codes to avoid any privacy concerns.

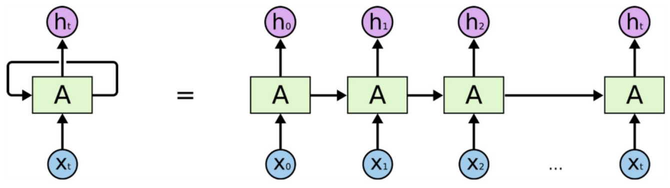
The dataset was downloaded from covid-19 nlp-text classification of Kaggle, which contained already made training and test datas. Only columns concerning the actual text sentence & label associated with it were used for training - since the rest contained location, time etc which were not of importance. Analyse the Sentiment and Classify the tweets. And predict which location , time have an upward trend in covid related discussions.If people are feeling positive or negative about the current pandemic.

III. Proposed Methodology

A. Transformers

Developed to solve the problem of sequence transduction or neural machine translation, these Transformers can work for any kind of input sequence to an output sequence dataset.

Although Recurrent Neural Network (RNN) works the same way of looping itself to let the info persist, it faces problems of long term memory dependencies making it difficult for examples that have a considerable difference between words of importance. Even Long Short Term Memory (LSTM) faces this same problem as it too processes the sentence word-by-word. This makes Transformers work successfully as its CNN + attention model where the idea is that there is relevant information behind every word in a sentence.



An unrolled recurrent neural network

We used BERT - and it's different versions. The diagram of a simple BERT cell is shown below:

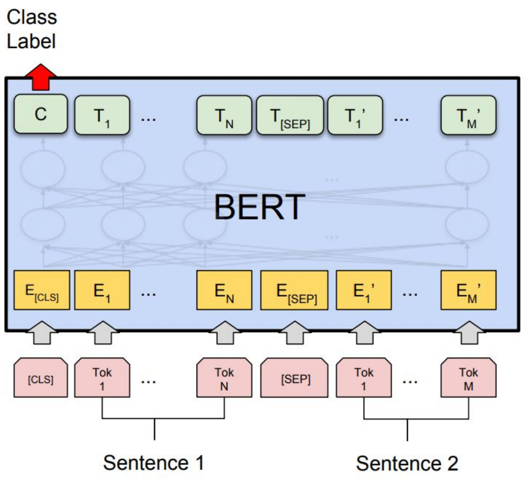


Photo attached shows a Transformer network - very different from the previous RNN network shown above.

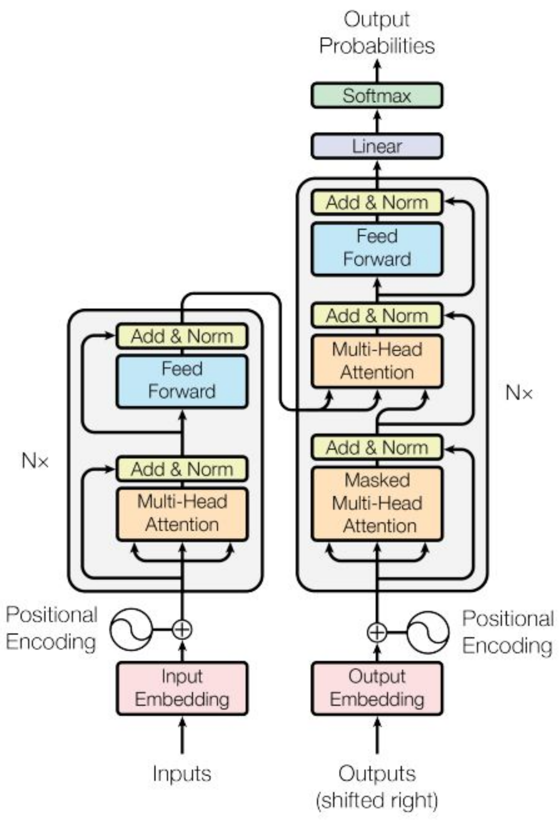


Figure 1: The Transformer - model architecture.

D. Training Inference Highlights

In our observation, The Maximum Speed Boost Was given By the novel Usage of FP16 DataType. It essentially consumes half the memory than FP32 for almost same performance. However the Overall Boost Was 150% deviating from the expected 200%.

We can see that training this model on a k80 Nvidia GPU gives us 80% accuracy. We trained it for 2 hours only, and we estimate that if we train it for 12 more hours the accuracy may go up to 85-90% . which we did.

E.Our Advantage And Superiority Over Others

Our novelty lies in the fact that we used AWS (Amazon Web Service) For our Program. We integrated S3 In Google Colab. Since Its Inception and integration with the cloud tech, it has enabled us to massively parallelize the training & running the 4 NLP models. This helped achieve a highest accuracy of 90% for bert, as well as Do A Transfer Learning From Other Checkpoints .

Another advantage is that we can improve feedback times and draw conclusions faster, By Spinning Up Multiple Instances and Synchronizing them with AWS. It Might Sound An Exaggeration , but the results are evident. By Our Programming Paradigm , Training Times And Code Setup Times Have Only Improved. The Improvement Can Be attributed to the fact that , We Do Not need to Setup And mount Google Drive Every Time We need to load a model, nor the Model State Disappears After every Session Termination, (session Terminations Are Very Common Problem With Free Tier Colab Users) . And Pro Service with fewer imitations is available only in The USA. Thus Students like us Have no choice but to develop their own Hybrid Solutions.

IV. COMPARING MODELS

We Trained 4 Models [2] With Different Parameters likeLayers, Hidden Layers, Heads , Batch Size Etc.We used the BERT Model, Upon which other models are developed. BERT Was initially Released in Nov 2018[3]

● Model: RoBERTa:

12-layer, 768-hidden, 12-heads, 125M parameters. RoBERTa using the BERT-base architecture

● Model: RoBERTa-Distilled:

6-layer, 768-hidden, 12-heads, 82M parameters. The DistilRoBERTa model distilled from the RoBERTa model roberta-base checkpoint.

● Model: XLNet-base:

12-layer, 768-hidden, 12-heads, 110M parameters. XLNet English model

● Model: XLNet-Large:

24-layer, 1024-hidden, 16-heads, 340M parameters. XLNet Large English model

After An Extensive reading From [4] [5] We Came to conclusion that we need to test both XLNet And BERT variants.

Abstract of XLNet :

The XLNet model was proposed in XLNet: Generalized Autoregressive Pretraining for Language Understanding by Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, Quoc V. Le. XLnet is an extension of the Transformer-XL model pre-trained using an autoregressive method to learn bidirectional contexts by *maximizing the expected likelihood over all permutations of the input sequence factorization order.*

Abstract of BERT:

The BERT model was proposed in BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding by Jacob Devlin, Ming-Wei Chang, Kenton Lee and Kristina Toutanova. It's a bidirectional transformer pretrained using a combination of masked language modeling objective and next sentence prediction on a large corpus comprising the Toronto Book Corpus and Wikipedia.

Abstract of DistillBERT:

The DistilBERT model was proposed in the blog post Smaller, faster, cheaper, lighter: Introducing DistilBERT, a distilled version of BERT, and the paper DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. DistilBERT is a small, fast, cheap and light Transformer model trained by distilling BERT base. It has 40% less parameters than bert-base-uncased, runs 60% faster while preserving over 95% of BERT's performances as measured on the GLUE language understanding benchmark.

The Gist of These Models And Their Performance Metrics Have been Logged On Github In the form of Jupyter Notebooks

V. CONCLUSION

This is our result for all the models that we've trained and used:

Models	RoBERTa -base	Distil RoBERTa	XLNet -base	XLNet -Large
Accuracy	91%	82.91 %	79%	75.92 %
Parameters	125M	82M	110M	340M
Training time	4 Hours	2.5 Hours	2 Hours	2.5 Hours

A. Conclusion can be made that the best model depending on no. of Parameters, time of Training as well as accuracy and error is RoBERTa.

B. Predicted and Actual results had all a difference of single class i.e. the most of the errors found was due to classification to its nearest label.

C. We further noted from this that the overall model can further be improved by combining or concatenating more than one model, though it'll surely increase the computational Cost exponentially.

D. Also the data was not perfectly labelled for some of the classes and were misclassified which didn't let the accuracy increase further.

E. DistillRobert Quickly Gained Accuracy then plateaued. The saturation was quick because parameters were less an there was nothing new left to learn. The accuracy climbed quickly as the parameters were less.

ACKNOWLEDGMENT

We extend our sincerest gratitude to Prof. T.V Vijay Kumar and Prof. Prerna Mukherjee for their constant support and guidance for this project.

REFERENCES

[1] https://www.kaggle.com/datatattle/covid-19-nlp-text-classification

[2]

https://huggingface.co/transformers/pretrained_models.html

[3]

https://ai.googleblog.com/2018/11/open-sourcing-bert-state-of-art-pre.html

[4]

https://www.kdnuggets.com/2019/09/bert-roberta-distilbert-xlnet-one-use.html

[5] arXiv:1906.08237

https://arxiv.org/abs/1906.08237

ABOUT BERT (Fictional Character):

Bert is a yellow Muppet character on the long running PBS and HBO children's television show Sesame Street. Bert was originally performed by Frank Oz. Since 1997, Muppeteer Eric Jacobson has been phased in as Bert's primary performer.