Opinion Mining Of Spanglish And Hinglish Text Data Using BERT And Transformer Model

Aviral Bajpai SRIP-2020

Summer Reseaerch Intern

Domain: Language and Dialogue Processing

SRIP
Report

Center for Cognitive Computing, IIIT-Allahabad

UNDER THE
SUPERVISION OF Prof. U.
S. Tiwari Professor
IIIT-ALLAHABAD

Table Of Contents

TOPIC

Opinion mining has been ordinarily connected with the examination of a content string to decide if a corpus is of a negative or positive sentiment. As of late, opinion mining has been stretched out to address issues, for example, recognizing objective from subjective suggestions and deciding the sources and points of various suppositions communicated in text informational collections, for example, tweets, message boards, web blogs, movie reviews, and news. Companies can use sentiment extremity and opinion point acknowledgment to pick up a more profound comprehension and the general extent of estimations.

Bidirectional Encoder Representations from Transformers (BERT) is a technique for NLP pre-training developed by Google.BERT makes use of Transformer, an attention mechanism that learns contextual relations between words (or sub-words) in a text.The attention mechanism was born to help memorize long source sentences in neural machine translation(NMT).The secret sauce invented by attention is to create shortcuts between the context vector and the entire source input. In its vanilla form, Transformer includes two separate mechanisms — an encoder that reads the text input and a decoder that produces a prediction for the task.Statistics show that half of the messages on Twitter are in a language other than English.

This evidence suggests that other languages, including multilingualism and code-mixing, need to be considered by the NLP community. The pre-trained model on massive datasets enables anyone building natural language processing to use this free powerhouse. BERT theoretically allows us to smash multiple benchmarks with minimal task-specific fine-tuning. This BERT model will be used with the code mixed tweets dataset which will help us get a better insight with the sentiment analysis task.

Dataset

The Dataset has been derived from an old Codalb's Competition "competitions.codalab.org".

From the competition named ,"SentiMix: Sentiment Analysis for Code-Mixed Social Media Text".

The objective of this dataset proposal is to bring the attention of the research community towards the task of sentiment analysis in code-mixed social media text.

Training Dataset for Hinglish and Spanglish data

1. https://ritual-uh.github.io/sentimix2020/data/hinglish_trial.txt

2. https://ritual-uh.github.io/sentimix2020/data/spanglish_trial.tx

Literature Survey

S.no	Title	Objective	Methods	Challenges Dealt
1.	BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding	Pre-train deep bidirectional representations from unlabeled text	Jointly conditioning on both left and right context in all layers.	The pre-trained BERT model can fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks.
2.	SentiBERT: A Transferable Transformer-Based Architecture for Compositional Sentiment Semantics	Proposal of SentiBERT, a variant of BERT that effectively captures compositional sentiment semantics	Incorporating contextualized representation with binary constituency parse tree to capture semantic composition.	Improvement in capturing negation and the contrastive relation and model the compositional sentiment semantics.
3	Attention Is All You Need	Proposal of a new simple network architecture, the Transformer	Attention mechanisms, dispensing with recurrence and convolutions.	Superior in quality while being more parallelizable and requiring significantly less time to train.
4	How to Fine-Tune BERT for Text Classification?	Experiments to investigate different fine-tuning methods of BERT on text classification task	Providing better performance on target task using models via transfer learning.	The proposed solution obtains new state-of-the-art results on eight widely-studied text classification datasets.

Work Plan

- Week 1-**2**: Learn the prerequisites for Transformers and Self-Attention to implement BERT model using PyTorch/TensorFlow framework.

- Week 3-**4**:To pre-process the data to make it ready for the model. Understanding the constraints and proceedING for the implementation .

- Week 5-6: Calibrate the hyper-parameters for a better model. Finding out more constraints to optimize the algorithm

- Week 7-8: Modification of the model for improvement of the results and writing the report.

Methodology

USING BERT MODEL

BERT (Bidirectional Encoder Representations from Transformers) is a recent paper published by researchers at Google AI Language. It has caused a stir in the Machine Learning community by presenting state-of-the-art results in a wide variety of NLP tasks, including Question Answering (SQuAD), Natural Language Inference (MNLI), and others.

BERT’s key technical innovation is applying the bidirectional training of Transformer, a popular attention model, to language modelling.

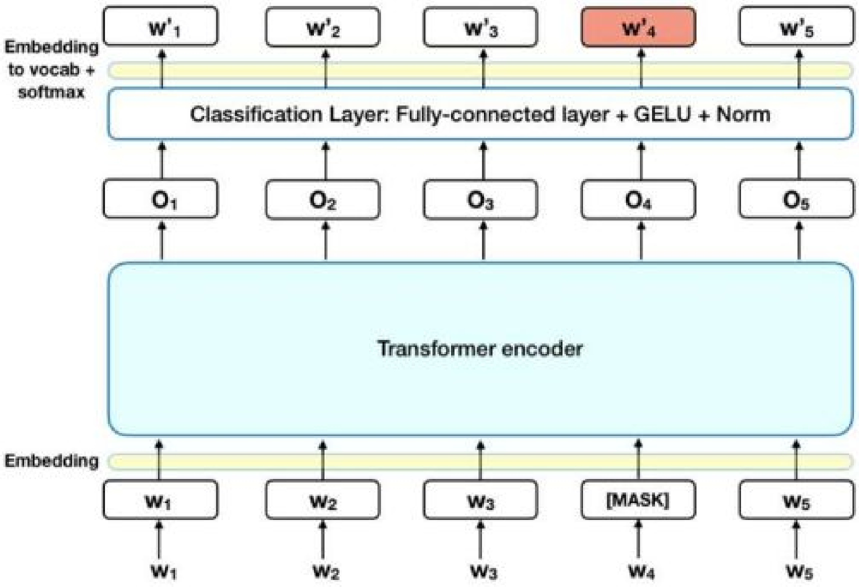
BERT Working BERT makes use of Transformer, an attention mechanism that learns contextual relations between words (or sub-words) in a text. In its vanilla form, Transformer includes two separate mechanisms — an encoder that reads the text input and a decoder that produces a prediction for the task.

Since BERT’s goal is to generate a language model, only the encoder mechanism is necessary. The detailed workings of Transformer are described in a paper by Google.

As opposed to directional models, which read the text input sequentially (left-to-right or right to-left), the Transformer encoder reads the entire sequence of words at once.

Therefore, it is considered bidirectional, though it would be more accurate to say that it’s non-directional. This characteristic allows the model to learn the context of a word based on all of its surroundings (left and right of the word).

The chart below is a high-level description of the Transformer encoder. The input is a sequence of tokens, which are first embedded into vectors and then processed in the neural network.

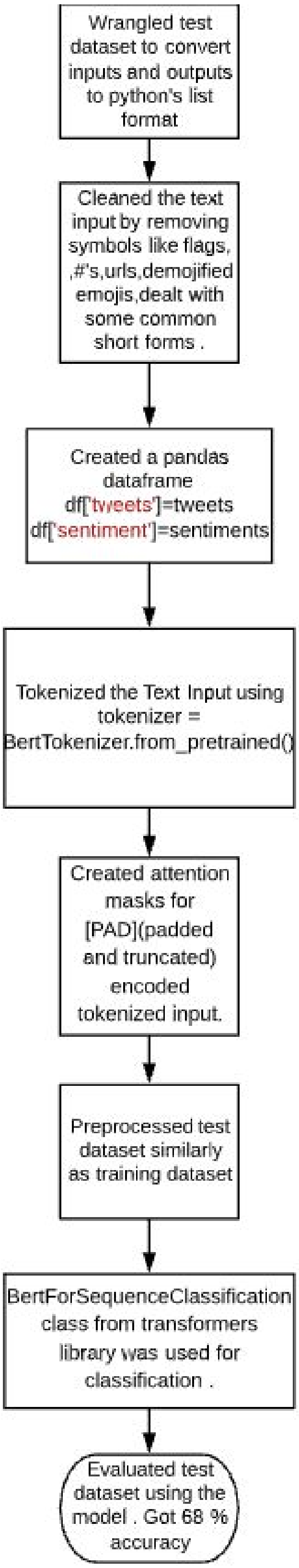


Embedding to vocab + softmax
w'1
w'2
w'3
w'4
w'5
Classification Layer: Fully-connected layer + GELU + Norm
O1
O2
O3
O4
O5
Transformer encoder
Embedding
w1
w2
w3
[MASK]
w5
w1
w2
w3
w4
w5

The output is a sequence of vectors of size H, in which each vector corresponds to an input token with the same index.

When training language models, there is a challenge of defining a prediction goal. Many models predict the next word in a sequence (e.g. “The child came home from ___”), a directional approach which inherently limits context learning.

Flow Chart Of Current Model


Wrangled test dataset to convert inputs and outputs to python's list format
Cleaned the text input by removing symbols like flags, ,#'s,urls,demojified emojis,dealt with some common short forms .
Created a pandas dataframe df['tweets']=tweets df['sentiment']=sentiments
Tokenized the Text Input using tokenizer = BertTokenizer.from_pretrained()
Created attention masks for [PAD](padded and truncated) encoded tokenized input.
Preprocessed test dataset similarly as training dataset
BertForSequenceClassification class from transformers library was used for classification .
Evaluated test dataset using the model . Got 68 % accuracy

Software And Tools Used

1. Google Colab

2. CUDA support

3. Colab Gpu Support

4. Pytorch(Framework)

5. Python(Language)

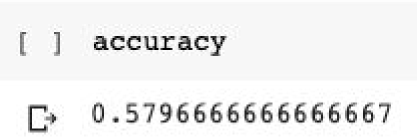
6. Libraries used -
    - transformers
    - matplotlib
    - re
    - numpy
    - pandas
    - sklearn
    - emoji

Results

The accuracy score of three models were evaluated on English-Hindi-Spanish multilingual test data .

The accuracies were found to be -

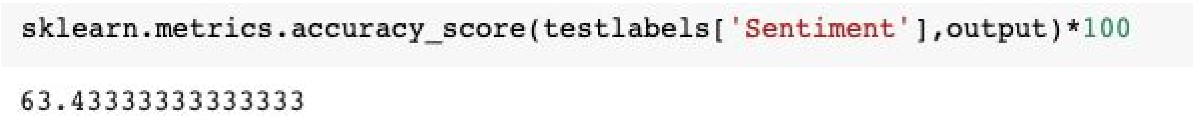
1. Bag Of Words Multi Layer Neural Network -



```
[ ] accuracy

0.5796666666666667
```

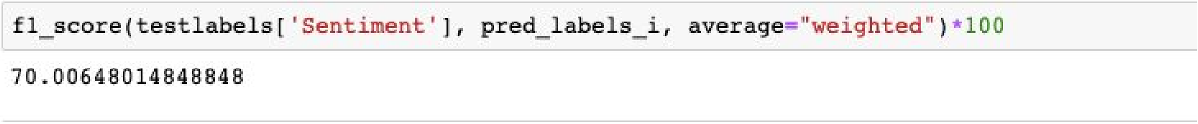
2. Sequence2Sequence LSTM model -



```
sklearn.metrics.accuracy_score(testlabels['Sentiment'],output)*100

63.43333333333333
```

3. Pretrained BERT Model -



```
f1_score(testlabels['Sentiment'], pred_labels_i, average="weighted")*100

70.00648014848848
```

It was very evident from Accuracies of each model on the test data that the pretrained BERT models performance was much better in mining the opinion of multilingual text data .

FUTURE SCOPE OF PROJECT

The accuracy of our model is 70% for our multilingual tweet dataset. The model can further be improved with more feature engineering tasks. This model scores much more on other standard datasets .

References

1. "Xin Li", "Lidong Bing", "Wenxuan Zhang", "Wai Lam" : Exploiting BERT for End-to-End Aspect-based Sentiment Analysis.
2. "Ashish Vaswani", "Noam Shazeer", "Niki Parmar", "Jakob Uszkoreit", "Llion Jones", "Aidan N. Gomez", "Lukasz Kaiser", "Illia Polosukhin" : Attention Is All You Need
3. "Jacob Devlin", "Ming-Wei Chang", "Kenton Lee", "Kristina Toutanova": BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding
4. "Chi Sun", "Xipeng Qiu", "Yige Xu", "Xuanjing Huang" : How to Fine-Tune BERT for Text Classification?
5. "Da Yin", "Tao Meng","Kai-Wei Chang" : SentiBERT A Transferable Transformer-Based Architecture for Compositional Sentiment Semantics
6. Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, Xuanjing Huang "Pre-trained Models for Natural Language Processing: A Survey" https://arxiv.org/abs/2003.08271
7. Da Yin, Tao Meng, Kai-Wei Chang "SentiBERT: A Transferable Transformer-Based Architecture for Compositional Sentiment Semantics" https://arxiv.org/abs/2005.04114