SENTIMENT CLASSIFICATION WITH CUSTOM NAMED ENTITY RECOGNITION

Ganga Babu M, Daniel.V, Bhagavathi Perumal,
Narendar Punithan, Pradeep Kumar S

Problem Definition

- When it comes to sentiment analysis challenges, there are quite a few things that companies struggle with in order to obtain sentiment analysis accuracy. Sentiment or emotion analysis can be difficult in natural language processing simply because machines have to be trained to analyze and understand emotions as a human brain does.
- While we typically look at emotion to capture feelings, such as anger, sadness, joy, fear, hesitation, sentiment is a higher-level classifier that divides the spectrum of emotions into positive, negative, and neutral. In our project we upgraded the spectrum to five set of emotions into Negative, Partially-Negative, Neutral, Partially-Positive & Positive.
- To build a classification model out of unprocessed text data/reviews using ML techniques to deal with information extraction from the real-world data/pipline on desired resources based on polarity of the reviews for sentiment analysis.

About Data Set

- Data are extracted from ‘Digital content and entertainment industry’ sourced from Data (cornell.edu)
- DATA: Movie Review Data extracted for the use of sentiment-analysis experiments which was orchestrated on 2002 at Cornell University, New York by Bo Pang & Lillian Lee.

Real Time Prediction Data Set

- Realtime data set is getting curated from our own choice such as Yelp & IMDB.

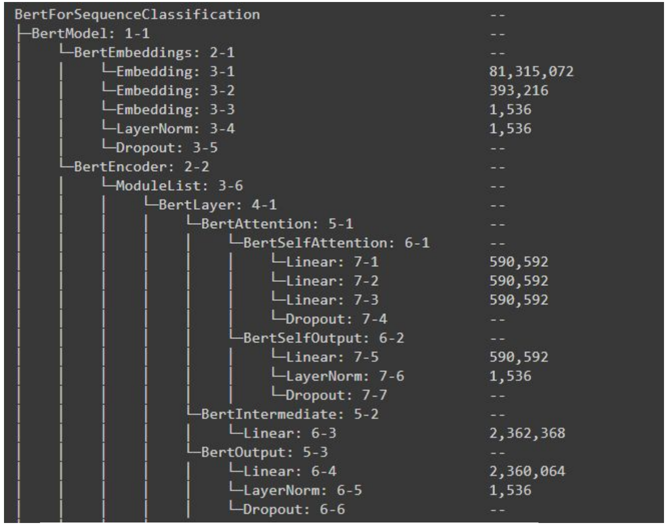
Suggested Solution & EDA

Solution:

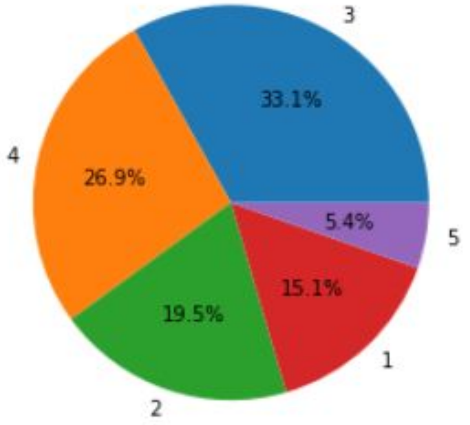
- Cleanse the Data through web scrapping process with the help of BeautifulSoup & Regular Expression Libraries.
- Pass the real time data to sentiment prediction Pipeline and get prediction from Built Model prepared from IMDB Movie review data

Sentiment Tokenizing

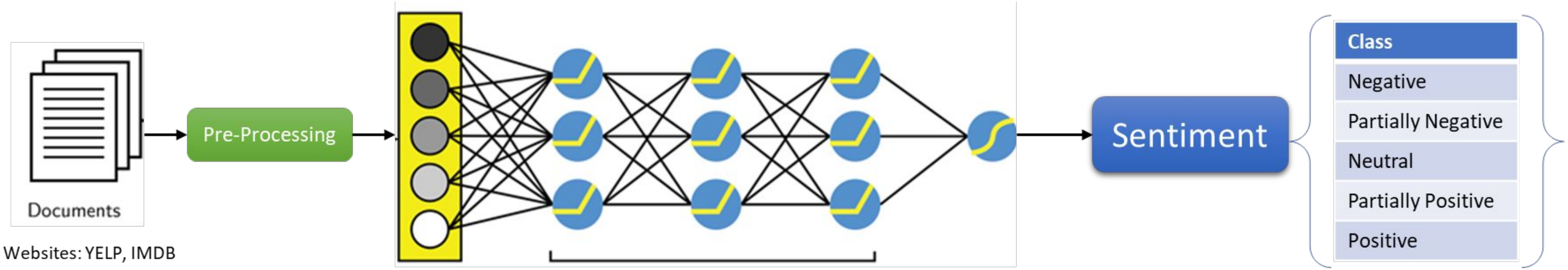
- With the help of Hugging Face Bert AutoTokenizer which comprises a BertForSequenceClassification of 12 BERT Layers which are BertSelfAttention mechanism a combination of Autoencoder and feedbacks.
- It took three hours to auto label each and every review for a corpus of 27K reviews, Such a way polarities are created for Movie Dataset.
- Built Bert model by onecycle policy with max lr of 2e-05
- Trainable params: 109,476,869 / Total params: 109,476,869
- Size of the model build tf_model.h5: 1314.47 MB



```
BertForSequenceClassification                           --
├─BertModel: 1-1                                        --
│    └─BertEmbeddings: 2-1                              --
│    │    └─Embedding: 3-1                              81,315,072
│    │    └─Embedding: 3-2                              393,216
│    │    └─Embedding: 3-3                              1,536
│    │    └─LayerNorm: 3-4                              1,536
│    │    └─Dropout: 3-5                                --
│    └─BertEncoder: 2-2                                 --
│    │    └─ModuleList: 3-6                             --
│    │    │    └─BertLayer: 4-1                         --
│    │    │    │    └─BertAttention: 5-1                --
│    │    │    │    │    └─BertSelfAttention: 6-1       --
│    │    │    │    │    │    └─Linear: 7-1             590,592
│    │    │    │    │    │    └─Linear: 7-2             590,592
│    │    │    │    │    │    └─Linear: 7-3             590,592
│    │    │    │    │    │    └─Dropout: 7-4            --
│    │    │    │    │    └─BertSelfOutput: 6-2          --
│    │    │    │    │    │    └─Linear: 7-5             590,592
│    │    │    │    │    │    └─LayerNorm: 7-6          1,536
│    │    │    │    │    │    └─Dropout: 7-7            --
│    │    │    │    └─BertIntermediate: 5-2             --
│    │    │    │    │    └─Linear: 6-3                  2,362,368
│    │    │    │    └─BertOutput: 5-3                   --
│    │    │    │    │    └─Linear: 6-4                  2,360,064
│    │    │    │    │    └─LayerNorm: 6-5               1,536
│    │    │    │    │    └─Dropout: 6-6                 --
```



Algorithm



Class
Negative
Partially Negative
Neutral
Partially Positive
Positive

Websites: YELP, IMDB

Dense Embeddings

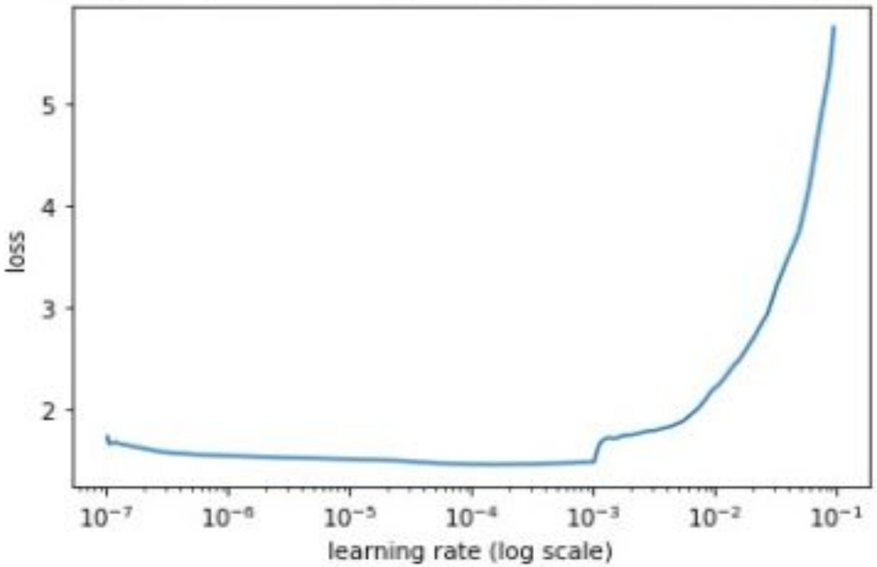
word2Vec, KMeans, Gensim,
Rule-based NLP, TextBlob vs VADER
BERT- AutoModel : HuggingFace
Auto-tokeniser & Ktrain

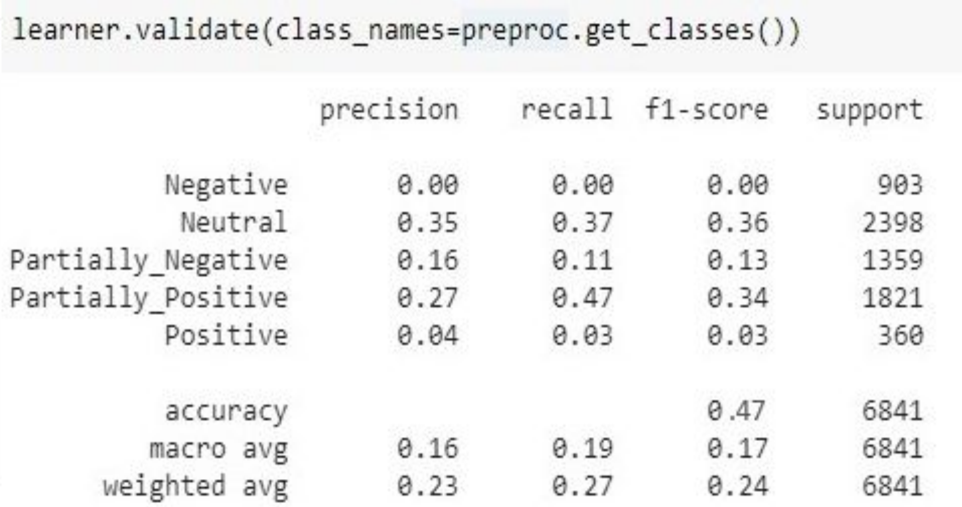
DNN -Hidden Layer

Bert – Pretrained Model

Bert – Pretrained Model

Model Evaluation



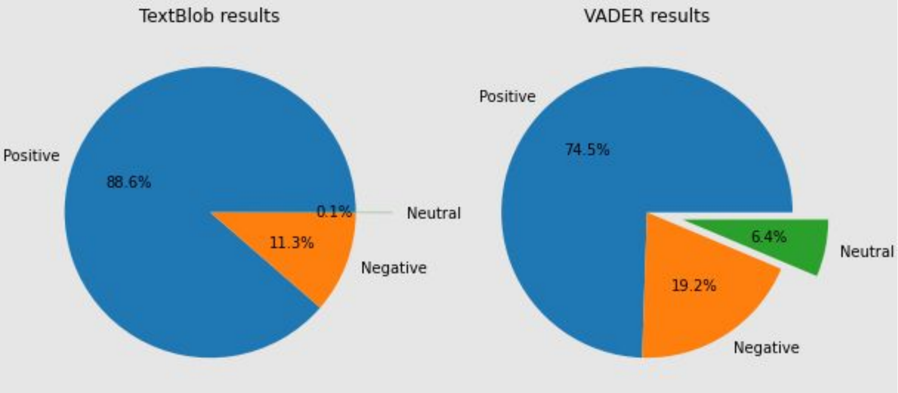


```
learner.validate(class_names=preproc.get_classes())

                    precision    recall  f1-score   support

          Negative       0.00      0.00      0.00       903
           Neutral       0.35      0.37      0.36      2398
Partially_Negative       0.16      0.11      0.13      1359
Partially_Positive       0.27      0.47      0.34      1821
          Positive       0.04      0.03      0.03       360

          accuracy                           0.47      6841
         macro avg       0.16      0.19      0.17      6841
      weighted avg       0.23      0.27      0.24      6841
```

Algorithms, Solution and Conclusions

Comparison to benchmark:

- Comparing performance to a benchmark definitely sets a higher “bar” than comparing to any other model experimented.
- TextBlob: Accuracy-83.98 | VADER: 80.29% | Ktrain Hugging face AutoTokenizer: 47%
- Even though TextBlob and Vader sets higher bar both failed during prediction. TextBlob failed to provide 0.1% of neutral sentiments when compared to TextBlob Vader was good.

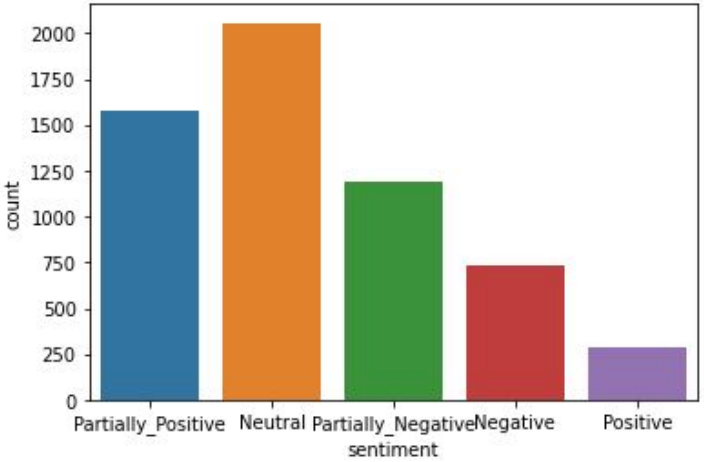


Algorithms, Solution and Conclusions

Amount of data fed during labelling

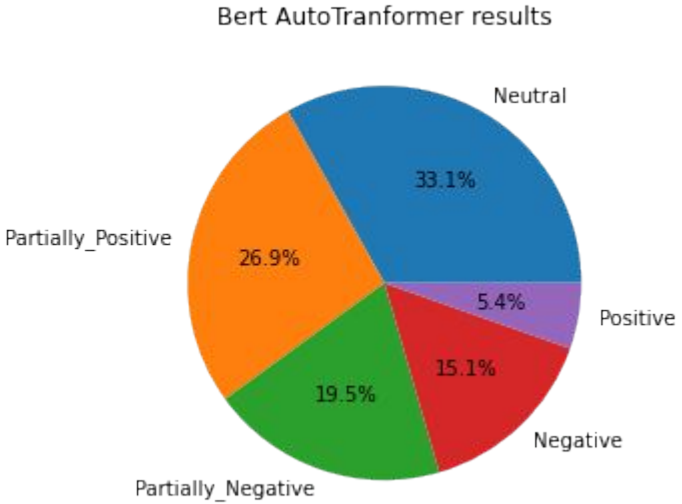
Neutral	2055
Partially_Positive	1577
Partially_Negative	1189
Negative	736
Positive	284

In Total 22000 for Training purpose



Prediction on final Model:

Sentiment Predicted on real-time data from IMDB movie: The suicide squad 2021



Thank You