THE GEORGE WASHINGTON UNIVERSITY

WASHINGTON, DC

6203: Machine Learning 2

Final Project Report – Group1

Nitin Godi

December 5th,2021

Table of Contents

Introduction

In this day and age, feedbacks and reviews from the customers dictates majority of the executive decisions, business strategies, etc. They also affect the stock prices of the companies. One such company is Amazon. Hence, feedbacks and reviews are rendered very important for the companies. However, analyzing the feedbacks and reviews is no easy task.

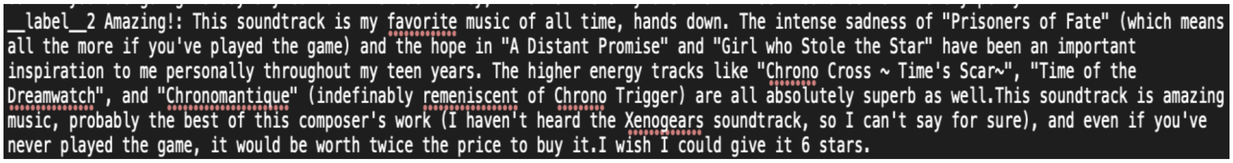
First and foremost, the shear amount of feedback and reviews collected by Amazon is humongous. Analyzing all the reviews and feedbacks is humanly impossible. Second obstacle in analyzing feedbacks and reviews is that they are in the form of text. They cannot be directly dumped into machine learning models. They need to be converted to numerical data before they can be analyzed. Third, traditional machine learning algorithms cannot handle huge amounts of data. Hence, the solution to this problem is neural networks.

The problem statement is to classify the reviews collected by Amazon based on their sentiments. This will help us understand the sentiments of the customers.

Data Description

The dataset used for the project is “Amazon Reviews for Sentiment Analysis”[1] obtained from Kaggle. This dataset contains reviews and their corresponding sentiments. This dataset will help us develop a model that can classify reviews accurately.

The dataset has two text files namely, ‘train.ft.txt’ and ‘test.ft.txt’. The train and test datasets are in fastText format. Figure 1 shows an example from the training dataset.



```
__label__2 Amazing!: This soundtrack is my favorite music of all time, hands down. The intense sadness of "Prisoners of Fate" (which means
all the more if you've played the game) and the hope in "A Distant Promise" and "Girl who Stole the Star" have been an important
inspiration to me personally throughout my teen years. The higher energy tracks like "Chrono Cross ~ Time's Scar~", "Time of the
Dreamwatch", and "Chronomantique" (indefinably remeniscent of Chrono Trigger) are all absolutely superb as well.This soundtrack is amazing
music, probably the best of this composer's work (I haven't heard the Xenogears soundtrack, so I can't say for sure), and even if you've
never played the game, it would be worth twice the price to buy it.I wish I could give it 6 stars.
```

Figure 1: Example of observations in dataset.

Each observation has the following format, <label> <title>: <review>. These observations were extracted and stored in the form of pandas data frame using regular expressions. The data frame has 3599330 observations and two columns namely, ‘label’ and ‘text’.

Column	Datatype	Description
text	object	Customer review
label	int64	0: Negative sentiment, 1: Positive sentiment

Algorithms

In this project, I have used three models. They are Mobile Bert, Distil Roberta and an Ensemble of Mobile Bert and Distil Roberta.

Mobile Bert

Basically, Mobile Bert model is a pre-trained model which made its first appearance in a research paper named “MobileBERT: a Compact Task-Agnostic BERT for Resource-Limited Devices” published by Zhiqing Sun, Hongkun Yu, Xiaodan Song, Renjie Liu, Yiming Yang, and Denny Zhou. It is a bidirectional transformer which is a compressed and accelerated version of BERT model.

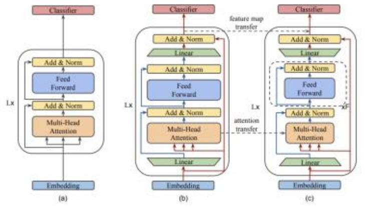


Figure 1: Illustration of three models: (a) BERT; (b) Inverted-Bottleneck BERT (IB-BERT); and (c) MobileBERT. In (b) and (c), red lines denote inter-block flows while blue lines intra-block flows. MobileBERT is trained by layer-to-layer imitating IB-BERT.

Figure 2: Mobile Bert Architecture[2].

Figure 2 shows the architecture of Mobile Bert.

Distil Roberta

Distill Roberta is also a pre-trained model which is a distilled version of Roberta model. Its training procedure is similar to Distil Bert. The model comprises of 6 layers, 768 dimensions and 12 head resulting in 82million parameters. This model was developed for English language and is case sensitive. It is modeled using Masked Language Modeling technique. Roberta was first introduced in the research paper “RoBERTa: A Robustly Optimized BERT Pretraining Approach” published by Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, Veselin Stoyanov.

Experimental Setup

This project was conducted on three different models. All the following steps are same for all the models except for modeling sections.

Model Training

Preprocessing

First the training dataset was loaded. Since the dataset was is fastText format, the dataset was converted into data frame using regular expressions. The dataset was then divided into training set and validation set.

Next, all the texts in training and validation sets were tokenized using tokenizer corresponding to the pretrained model used. After tokenization, data loaders for both the sets were created.

Modeling

For Mobile Bert model, MobileBertForSequenceClassification was used from 'google/mobilebert-uncased' pre-trained model.

For Distil Roberta model, RobertaForSequenceClassification was used from 'distilroberta-base' pre-trained model.

For the Ensemble model, the outputs of the models mentioned above were passed to a Linear layer with input dimension of 4 and output dimension 2.

Training

First the optimizer, scheduler and criterion were set. The optimizer used in the project is AdamW and the criterion used in the project is Cross Entropy Loss.

Next, the model was put in train mode and outputs & losses were calculated using batches of data. Using the loses the weights and biases of the model were updated.

The model was then put I evaluation mode. Then, the outputs were generated for validation data.

Next, accuracy, precision and recall scores were calculated on the outputs of validation data.

Finally, the weights and biases of the model were stored.

Testing

In testing phase, the test data was preprocessed and tokenized. Then, the model used for training was developed and its weights and biases were loaded. Lastly, the outputs for the test data were generated and the accuracy, precision and recall scores for these outputs were calculated.

Results

This experiment was performed using different pre-trained models. The table below shows their performances. These pre-trained models did not as good as the Mobile Bert model, Distil Roberta model and the ensemble model containing Mobile Bert and Distil Roberta models.

Models	Accuracy	Precision	Recall
Distil Bert	0.9291	0.9289	0.9273
Electra	0.9323	0.9316	0.9269
Distil Roberta	0.9346	0.9350	0.9307
Mobile Bert	0.9296	0.9308	0.9283
Flaubert	0.8778	0.8921	0.876
Ensemble of Electra and Distil Bert	0.92894	0.9377	0.92373
Ensemble of Electra and Electra	0.92773	0.92568	0.93436
Ensemble of Mobile Bert and Electra	0.93487	0.93672	0.93655
Ensemble of Mobile Bert and Distil Roberta	0.94080	0.9462	0.9378
Ensemble of Electra and Distil Roberta	0.89364	0.89047	0.90275
Ensemble of Mobile Bert and Distil Bert	0.93199	0.93633	0.93197

According to the metrics calculated while using different models, the best model is the ensemble of Mobile Bert and Distil Roberta models. The table below shows the metrics calculated on test data for different models.

Models	**Accuracy**	**Precision**	**Recall**
Mobile Bert	0.9296	0.9302	0.9283
Distil Roberta	0.9346	0.9350	0.9307
Ensemble of Mobile Bert and Distil Roberta	0.94080	0.9462	0.9378

In the Kaggle dataset description of the dataset, precision and recall values using fastText method is said to be 0.916. However, all the three models used in this project outperformed the fastText method. From the table above you can see that all three models have precision and recall values more than 0.92. Out of the three models mentioned above, the ensemble model performed the best with 94% accuracy, 94.6% precision, and 93.7% recall.

Summary and Conclusion

The main objective of the project was to train a model that best classifies the reviews using the dataset from Kaggle named, Amazon Reviews for Sentiment Analysis. The reviews were preprocessed, tokenized, fed into different models in order to train them and determine the best model.

The data was trained on by 11 different models. Out of which 9 models were able to classify reviews with precision and recall scores over 0.916 which was the benchmark mentioned in Kaggle dataset description. However, I chose three models to talk about in this project namely, Mobile Bert, Distil Roberta and the Ensemble of these two models.

In this project, pre-trained models were used directly without any major changes. Hence, in the future, these pre-trained models can be manipulated and used to try and find a much better model than the ones in this project.

References

- Kaggle dataset used
    - https://www.kaggle.com/bittlingmayer/amazonreviews
- Sun, Z., Yu, H., Song, X., Liu, R., Yang, Y., & Zhou, D. (2020). Mobilebert: A compact task-agnostic Bert for Resource-limited devices. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. https://doi.org/10.18653/v1/2020.acl-main.195
- Hugging face website for Mobile Bert
    - https://huggingface.co/docs/transformers/model_doc/mobilebert
- Hugging face website for Distil Bert
    - https://huggingface.co/distilroberta-base
- Base code to ensemble models
    - https://discuss.pytorch.org/t/custom-ensemble-approach/52024/4
- Base code structure
    - https://huggingface.co/docs/transformers/training#finetuning-in-native-pytorch