Module: INM706 Deep Learning 2: Prediction

Professor: Dr. Alex Ter-Sarkisov

Sentiment Prediction for Drugs Reviews using BERT

Authors:

Jasveen Kaur - 190020638

Elisabeta Monica Furdui - 190045971

1. Introduction

Online pharmaceutical drug reviews are a valuable source of information, not only to the patients, but also to the manufacturers and the sellers of the drug. Besides the doctor's recommendations, patients check online for reviews of the drugs they receive prescription for, and often write their own reviews to help other patients. Automatic sentiment prediction of patient reviews can be a direct source to improve quality of care and patient satisfaction (Korkontzelos et al., 2016). For example, expressing negative sentiments through online reviews, adverse reactions to drugs, can be identified, which otherwise may be missed in drug trials, due to the long term use of a drug that trials lack, combination of multiple drug usage, or the fact that several categories of patients are excluded from trials such as children and elderly (Korkontzelos et al., 2016). At the same time, pharmaceutical drug manufacturers can take into consideration constructive criticism to improve the quality of their drugs, whereas the drug selling companies can increase their supply of the drugs that have positive reviews. As such sentiment prediction is an important task for pharmaceutical drug reviews. This project aims to predict the sentiment of drug reviews using the BERT (Bidirectional Encoder Representations from Transformer) model.

Conventional Language models such as RNN and LSTM have been effective enough, however, they have certain limitations when the length of input sentences is long. RNN can’t learn long-term dependencies across different time steps. Due to its short-term memory it suffers from the vanishing/exploding gradient problem. This problem could be overcome by LSTMs, however LSTMs cannot incorporate very long sequences since they follow a sequential path. At the same time, training LSTMs is slow because they do not make use of GPUs designed for parallel computations.

The above-mentioned drawbacks can be overcome with the use of Transformers (Vaswani et al., 2017). Transformers used the attention mechanism, which captures long term dependencies based on the importance of each word in the language model rather than the whole sentence. BERT (Bidirectional Encoder Representations from Transformer) is also a transformer model, which incorporates the self-attention mechanism. It has achieved State-of-the-Art results on many language modelling tasks such as text classification, next sentence prediction and question-answer data.

Dataset Interface

The sentiment prediction task is performed on drug reviews dataset (Gräßer et al., 2018), taken from UCL Machine Learning repository. The dataset contains pharmaceutical drug user reviews collected through crawling methods from two websites, Drug.com and Druglib.com. The dataset contains 7 columns (as in Figure 1.)

1.Unnamed:0 (id)

2.drugName

3.condition (the illness that the reviewer claims to have)

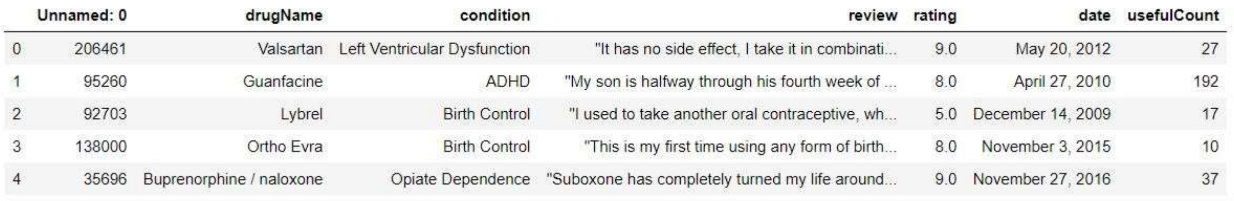
4.review (the text containing the review)

5.rating (10 star rating, 0 being lowest and 10 being highest)

6.date (the date when the review was submitted)

7.usefulCount (the number of people that voted the review as a useful one)

It consists of 215063 drug reviews of 3436 unique drugs given by 161297 users. The dataset is divided into a training set and a testing set, consisting of 161297 and 53766 rows respectively. The .tsv format files were first loaded as dataframes and then preprocessed.



	Unnamed: 0	drugName	condition	review	rating	date	usefulCount
0	206461	Valsartan	Left Ventricular Dysfunction	"It has no side effect, I take it in combinati...	9.0	May 20, 2012	27
1	95260	Guanfacine	ADHD	"My son is halfway through his fourth week of ...	8.0	April 27, 2010	192
2	92703	Lybrel	Birth Control	"I used to take another oral contraceptive, wh...	5.0	December 14, 2009	17
3	138000	Ortho Evra	Birth Control	"This is my first time using any form of birth...	8.0	November 3, 2015	10
4	35696	Buprenorphine / naloxone	Opiate Dependence	"Suboxone has completely turned my life around...	9.0	November 27, 2016	37

Figure 1. Raw Dataframe example

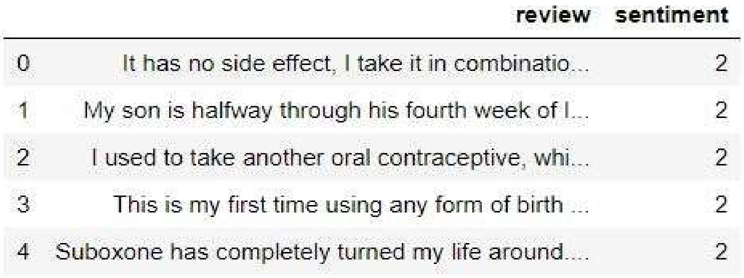
Sample review before preprocessing

‘ “I had Crohn&#039;s with a resection 30 years ago and have been mostly in remission since. Have recently had a bad flare and narrowing at the anastomosis and need to be on medication, but haven&#039;t found one that I can handle. Asacol gave me such serious body aching and fatigue that I could not function. Pentasa immediately gave me heart palpitations and arrhythmias so I had to discontinue it." ’

The datasets were first labelled as positive (label: 2) for the reviews where the rating is between and including 8 to 10, negative (label: 0) where the rating is between and including 0 to 3, and neutral (label: 1) where rating is between and including 4 to 7. The columns that were irrelevant to the sentiment prediction were dropped such as drugName, condition, date, usefulCount, Unnamed: 0. A helper function was defined to preprocess the reviews and remove special characters such as @,& etc using regex library.

Sample review after preprocessing

"I had Crohn's with a resection 30 years ago and have been mostly in remission since. Have recently had a bad flare and narrowing at the anastomosis and need to be on medication, but haven't found one that I can handle. Asacol gave me such serious body aching and fatigue that I could not function. Pentasa immediately gave me heart palpitations and arrhythmias so I had to discontinue it."



	review	sentiment
0	It has no side effect, I take it in combinatio...	2
1	My son is halfway through his fourth week of I...	2
2	I used to take another oral contraceptive, whi...	2
3	This is my first time using any form of birth ...	2
4	Suboxone has completely turned my life around....	2

Figure 2. Dataframe after preprocessing

Further Bert specific preprocessing and data loading using Pytorch was performed. It involves loading pre-trained Bert tokenizer (bert-base-cased and Biobert) and then tokenizing the dataset using tokenizer.encode_plus function. Special tokens, attention masks and input ids were added. The reviews were converted to ids and truncated to a maximum length of bert tokenizer i.e. 512 and the masks and ids were returned as pytorch tensors.

Special tokens:

Token	Token ID	Description
[UNK]	100	Used as ids for the words which do not belong to the bert training set i.e. unknown
[SEP]	102	Added at the end of each sentence to mark the ending of that sentence
[CLS]	101	It is added at the start of each sentence for classification task
[PAD]	0	Used to fill the maximum sequence length in case of a review shorter than max_len

Table 1: Token IDs and description of special tokens

Attention Masks: They are used to indicate whether the input is a real token, with a mask of 1 or the token does not belong to the sequence, with a mask of 0.

Data Splitting: After Bert specific preprocessing, the training set was split into validation (20%) and train (80%) sets. The reviews under each class for the three sets are as shown below:

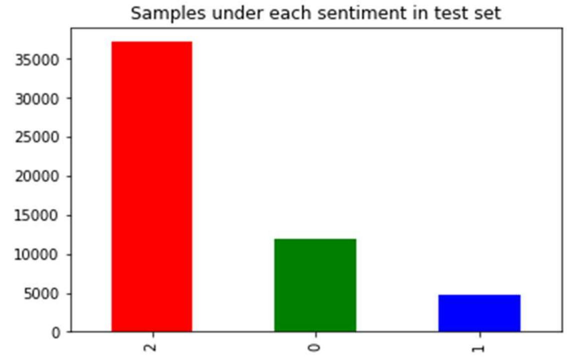


Figure 3. Test set

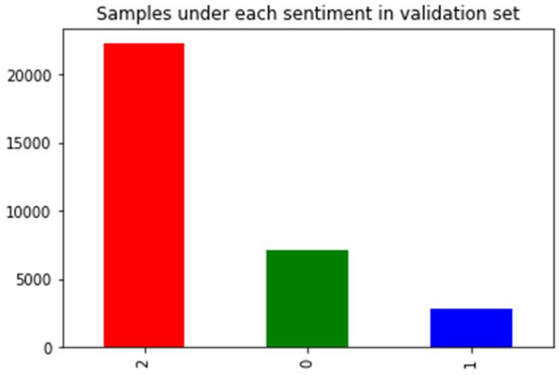


Figure 4. Validation set

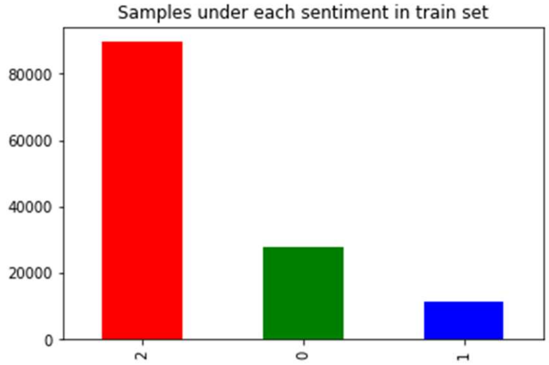


Figure 5. Train set

Dataloading: A function (create_data_loader) was defined, which used the Pytorch Dataloader class to load the dataset. The size of the dataset was reduced while loading. Since Bert is a pretrained model, it can be fine-tuned according to the given task even with fewer samples. It also decreased the training time of the model.

3. Model

The **transformer** model was introduced as a faster to train, simpler model, to RNN and LSTMs. Transformers are based only on self-attention mechanisms and they are deeply bidirectional (Vaswani et al., 2017). The input sequence for a transformer can be passed in parallel, unlike in LSTMs where the input is passed sequentially. Transformers incorporate the concept of attention i.e. focusing on the important words and leaving out the less important ones based on relevance. It uses six encoder and decoder layers, each followed by multi-head attention sublayer and feedforward network and at the end layer normalization followed by residual connection.

The input to the encoder is the input embedding, where the embedding space maps words to vectors and the positional encoder gives context to words based on their position in a sentence. The word vectors with the positional information are then passed to the encoder block where they go through a multi headed attention layer and feed forward layer. The multi-head attention block generates the attention vectors for each word, which are then passed through a feed forward layer one vector at a time, which allows parallelization. The decoder takes the output of the encoder into an additional masked encoder-decoder attention block (masked inputs in order to promote learning) and then feeds it to a multi-head attention block, a feed forward layer, a linear layer and lastly softmax. The softmax layer predicts the probability of the next word in the sequence.

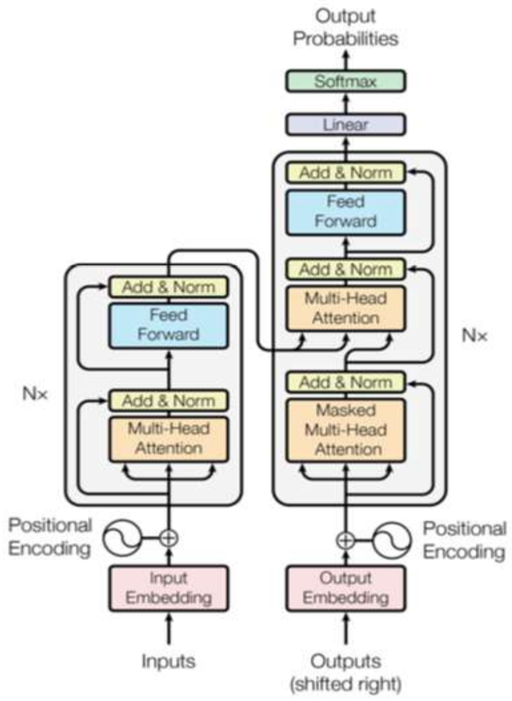


Figure 6. Transformer architecture

BERT (Bidirectional Encoder Representations from Transformer) is an unsupervised and deeply bidirectional pre-trained language model (Devlin et al., 2019) which has produced state-of-the-art results in many NLP tasks such as question and answering systems and sentiment analysis. The framework has 2 parts: Pre-training and Fine-tuning (Devlin et al., 2019). It has been pre-trained on English Wikipedia and BookCorpus. Pre-training on such a large dataset enabled it to learn more in-depth features of the language. Hence, it can be fine-tuned to achieve high accuracy on any NLP task, even with the availability of a small dataset. Previous models like ULMFit and ELMo are unidirectional and shallow bidirectional, in comparison to BERT, which is deeply bidirectional and therefore it can extract more context features. BERT was pre-trained using two unsupervised tasks:

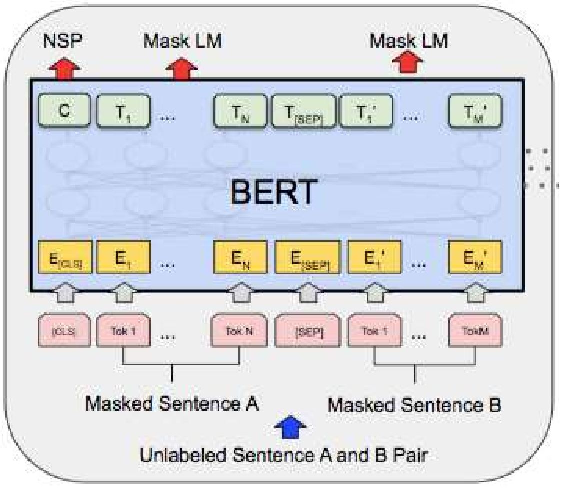


Figure 7. BERT Pre-train architecture (Devlin et al., 2019)

Masked ML and Next sentence Prediction (NSP) (Devlin et al., 2019). During the Masked LM procedure 15% of the tokens in a sentence are masked randomly and then run through the model to predict the masked token. Through this procedure the model learns bidirectional context. Because understanding relationships between two sentences is important for many NLP tasks such as question answering systems and natural pretrained on Next sentence Prediction (NSP). During NSP 50% of the time a sentence follows the next sentence, 50% of the time the next sentence is taken randomly from the corpus. By pretraining the model on NSP, the model learns sentence relationships needed for many NLP tasks. The next step in using BERT is fine-tuning the pre-trained model on a specific NLP task.

In terms of architecture, BERT has two variants:

1. BERT Base: uses 12 blocks of transformer encoder (layers), 12 attention heads and 110 million parameters.
2. BERT Large: uses 24 blocks of transformer encoder (layers), 16 attention heads and 340 million parameters.

BERT for Sentiment Prediction

For the purpose of this work, we fine-tuned general and domain specific Bert Base models on multiclass sentiment prediction task.

The tokens ids generated by passing the strings to the bert tokenizer, along with addition of special tokens in the sequence are fed into the bert model. The Bert (Base) model outputs a vector of hidden size i.e. 768. The output embedding is passed through a linear layer and then a softmax classifier. The softmax classifier outputs the probabilities of the input sequence belonging to each class.

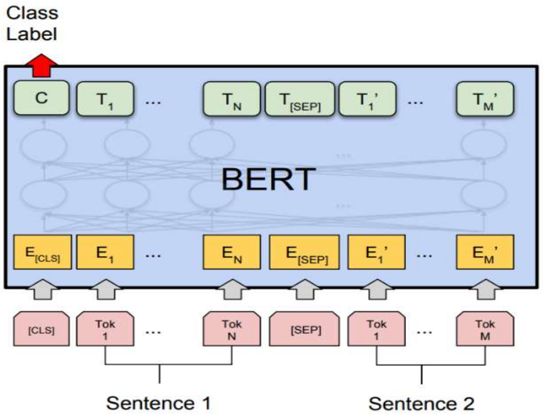


Figure 8. Bert for sentiment classification

The models used were imported using Transformers library as:

```
Import transformers
from transformers import BertModel, BertTokenizer
bert_model = BertModel.from_pretrained('bert-base-cased')
```

4. Custom Functionalities

Pre-trained models:

Due to the limitations of the BERT model on domain specific NLP tasks, bert was adapted on biomedical data to obtain better results on medical domain. Pretrained weights such as BioBert and Clinical Bert were introduced.

1. **“bert-base-cased"**: - version of BERT where accents are maintained, and it does not convert the words to lowercase.

Bert-base has the following architecture:
12-layer, 768-hidden, 12-heads, 110M parameters.

3. **BioBert** (Lee et al., 2019): It is a language representation model for biomedical text mining tasks. It was pre-trained on the original Bert model using biomedical corpora i.e. PubMed extracts and PMC full text articles consisting of 4.5 billion words and 13.5 billion words respectively. It largely improved the performance of the Bert model on various medical tasks. It has the same architecture as that of Bert.

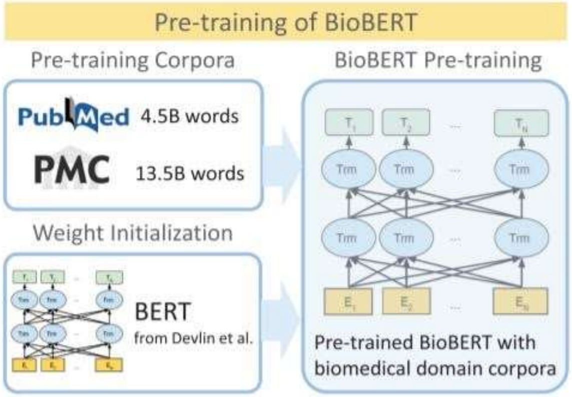


Figure 9. Pre-training BioBERT (Lee et al., 2019)

4. **BioClinicalBert** (Alsentzer et al., 2019): This model is initialized from BioBert and trained on physicians’ medical notes from MIMIC II database which contains electronic health records of ICU patients. Since it is an improvement on the pre-trained BioBert model, it is expected to perform better in the clinical domain.

Additional Layers

After loading the pretrained Bert, the following layers were added at the end as two separate models. Both the models were followed by a softmax classifier.

1. **Linear Layers**: Two linear layers: One hidden layer with 768 input nodes and 512 output nodes, followed by a ReLU nonlinearity and dropout with probability 0.3. The output layer consisted of 512 input nodes and 3 output nodes (for 3 classes i.e. sentiments).

2. **GRU Layers**: One bidirectional GRU layer layer was implemented on top of the bert model, followed by dropout with probability 0.2 and an output layer with 3 output nodes indicating the classes.

Optimizers:

1. **AdamW**: Decoupled weight decay regularization i.e. AdamW was used. It helps to improve Adam's generalization performance (Loshchilov and Hutter, 2019). It was used using Hugging Face's implementation, which uses fixed weight decay with the Adam algorithm.

2. **RMSprop**: It is Root Mean Square Propagation proposed by Geoffrey Hinton. Pytorch's implementation of the RMSprop algorithm was used for the experiment.

Loss function: Cross Entropy loss: It combines LogSoftMax with negative log likelihood loss. The Pytorch implementation uses the formula given below. Each minibatch loss is then averaged across all the observations.

$$loss\ (x, class)\ =\ -\,log\left(\frac{exp\ (x[class])}{\sum_j exp\ (x[j])}\right)$$

$$=\ -\,x[class]\ +\ log\left(\sum_j exp\ (x[j])\right)$$

Accuracy metrics:

1. **Accuracy**: It used to evaluate models and it is calculated by dividing the number of correct predictions with the number of total predictions. It was calculated using NumPy mathematical operations.

2. **F_1 Score**: It is a measure of a test dataset accuracy. It is calculated as a weighted average of the precision and recall of the test. An F_1 score =1 is the best and F_1 score=0 is the worst. The formula for F_1 is:

$$F_1 = 2 * (\text{precision} * \text{recall}) / (\text{precision} + \text{recall})$$

It was calculated using: sklearn.metrics.f1_score accuracy metric provided by scikit-learn library. It uses three types of average functions to calculate the F_1 score: weighted, micro and macro. Macro average does not take data imbalance into account, therefore, it gives low value.

5. Training

The training process involved fine-tuning the various BERT models described in section 3 on our drug reviews dataset. The Hugging Face transformers library was used to download the pretrained weights and the tokenizers for the training process for all the models described above. The inputs to the neural network were the input ids and attention masks of the training and validation set, generated by tokenizer.encode_plus (section 1) function.

For the training purpose, 10000 reviews from the train set and 8000 reviews from the validation set were taken. Fewer reviews reduced the training time. During training, the validation set was used to evaluate the model parameters. The batch size was kept fixed, i.e. 16. Since the maximum length of each review was 512, increasing the batch size gave a memory error. The learning rate was fixed i.e. 2e-5.

Additional Layer:

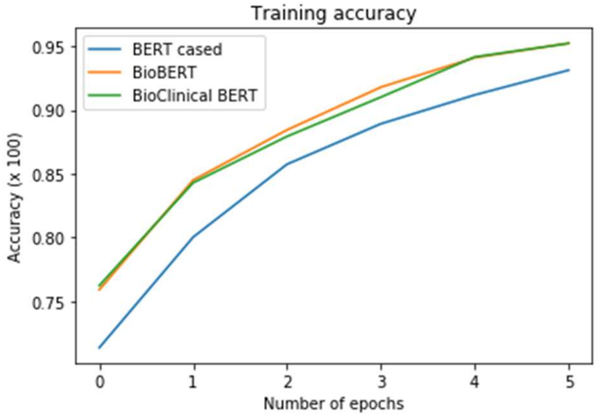
It was observed that the model with two linear layers performed better than the GRU layers. GRU layers gave accuracy of 76.92% with Bert base cased model. Whereas, Linear layers gave training accuracy of 93.11% at the end of 6 epochs.

Pre-trained Models:

	Epoch 1	Epoch 2	Epoch 3	Epoch 4	Epoch 5	Epoch 6
Bert Cased	71.37 (0.761)	80.03 (0.562)	85.74 (0.424)	88.91 (0.332)	91.17 (0.274)	93.11 (0.227)
BioBert 1.1	75.91 (0.646)	84.50 (0.447)	88.43 (0.330)	91.78 (0.252)	94.09 (0.201)	95.22 (0.170)
BioClinical Bert	76.24 (0.642)	84.29 (0.448)	87.92 (0.332)	91.00 (0.251)	94.15 (0.193)	95.21 (0.166)

Table 2. Training Accuracy (and loss)

The above table shows that BioBert and BioClinical Bert have maximum accuracy at the end of 6 epochs, therefore, they are well trained as compared to other models. Although the two domain-specific models (BioClinical Bert and BioBert) outperformed the Bert Cased model by a small margin, there was not a significantly large difference between the training accuracies and the loss values of the three models.



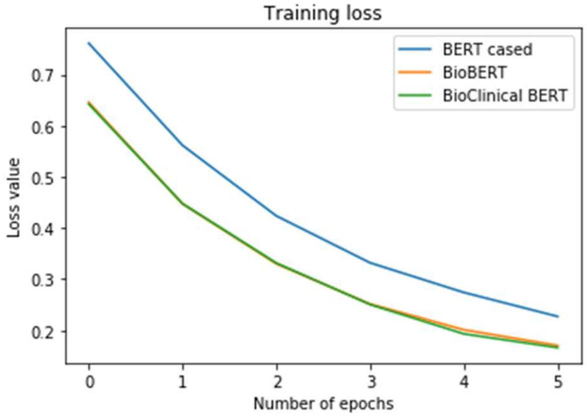


Figure 10. Training accuracies of different BERT models1

Figure 11. Training loss of different BERT models

Optimizer: The model giving the highest training accuracy from the pre-trained weights was used to compare the performance of AdamW optimizer and RMSprop. It was observed that RMSprop took longer time to train as compared to AdamW and the performance of the RMSprop model was also low as compared to AdamW.

6. Evaluation

The evaluation of the trained model was done using the accuracy metric, f1 score and loss value. It is also done by predicting the sentiment of individual sentences on the test dataset as well as raw reviews using the most efficient model i.e. with the model with highest training accuracy: “BioBert”.

Test Results:

Model	Test Accuracy (%)	Test Loss	F1 Score (weighted)	F1 Score (macro)	F1 Score (micro)
Bert Cased	80.59	0.623	0.779	0.560	0.806
BioBert 1.1	81.08	0.516	0.773	0.534	0.811
BioClinical Bert	80.40	0.594	0.764	0.523	0.804

Table 3: Test results on drug reviews dataset

The highest test accuracy was observed on BioBert model, which was comparable enough to Bert case and BioClinical Bert. However, its low F1 macro score is due to the highly imbalanced dataset (as shown in Fig. 3,4 and 5).

Sentiment prediction of the reviews in test data:

A function was defined to predict the sentiment values of the individual reviews in the test set, which returned the review text, predicted sentiment, probability of each sentiment and the true probabilities. A sample of sentences were displayed to check the sentiment predicted by the model.

Correct predictions:

Sample Review 1: *"They didn&#039;t help my dry eyes at all."*

Predicted sentiment value: 0	*True sentiment value: 0*
Predicted sentiment: negative	*True sentiment: negative*

Sample Review 2*: "My son has Crohn&#039;s disease and has done very well on the Asacol. He has no complaints and shows no side effects. He has taken as many as nine tablets per day at one time. I&#039;ve been very happy with the results, reducing his bouts of diarrhea drastically."*

Predicted sentiment value: 2	*True sentiment value: 2*
Predicted sentiment: positive	*True sentiment: positive*

The above examples show that our fine-tuned model successfully predicted the correct sentiment of the reviews in the test set for most of the reviews. However, there were instances where it failed to predict the neutral sentiment correctly because of the imbalanced dataset (fewer samples of neutral sentiment). This issue can be resolved by resampling the dataset with equal numbers of reviews in the three classes.

Model Failures:

Sample Review 1: *"Ive been on Methadone for over ten years and currently,I am trying to get off of this drug. Ive been decreasing my does 2 mgs per month for over a year. I am at 3 mgs and really starting to feel the withdraw.I don&#039;t plan to get my next 30 doses.because its almost rediculous how little it does for me. I have 3 does doses of 3 mg and Im terrified. Can anyone give me some truthful encouragement?....."*

Predicted sentiment value: 0	*True sentiment value: 1*
Predicted sentiment: negative	*True sentiment: neutral*

Sample Review 2: *"Have been on Actos for almost a year, gained 24 pounds and have swelling in hands and feet and are retaining a lot of water in my thighs. My sugar levels are good. My doctor lowered my dosage from 30 mg to 15 mg but refused to take me off. Will get a second opinion because the side effects are too much."*

Predicted sentiment value: 2 *True sentiment value: 1*
Predicted sentiment: positive *True sentiment: neutral*

Sentiment Prediction of raw reviews (not specifically medical domain):

Raw review refers to the sentences given as input that are not part of the given datasets, to check the application of our trained model. The sentences were first tokenized and processed using tokenizer.encode_plus to return the input ids and attention masks. The input ids and attention masks were used as input to the model. The output of the model was passed through the Softmax function to check the probability of the review belonging to the three classes. The maximum value of the probability depicts the predicted class (sentiment) of the review.

Sample Review 1: *"I need to take many dosages to see the effect. It harms my nervous system and makes me feel more sick. I am not taking it again and will not recommend it to anyone."*

Probabilities: (0.8954, 0.0632, 0.0414); Sentiment value: 0; Sentiment: negative

Sample Review 2: *"This medicine gives me headaches but is really effective in case of severe cardiovascular muscle pain. I hope they improve it in the future."*

Probabilities: (0.0259, 0.0867, 0.8874); Sentiment value: 2; Sentiment: positive

Sample Review 3: *"This is a great medicine. It would surely recommend it to anyone above the age of 18."*

Probabilities: (0.0101, 0.0303, 0.9596); Sentiment value: 2; Sentiment: positive

Our model is good at predicting positive and negative sentiments. Since the number of samples of neutral sentiments are too low, it fails to predict neutral sentiment in most of the cases. The training and test accuracy also indicate that the model performs well.

7. Comparison

Sentiment analysis on drug reviews performed by (Bised and Mo, 2020) experimented with 8 different BERT models. Their implementation used a batch size of 32 with 128 maximum sequence length. The following table shows our test accuracy on 1 epoch in comparison to their test accuracies at 1st epoch and at the end of 4th epoch.

	Bised and Mo, 2020 Epoch 1 (%)	Bised and Mo, 2020 Epoch 4 (%)	Our implementation (1 epoch) (%)
Bert Cased	82.4 (0.440)	88.8 (0.510)	80.59 (0.623)
BioBert	82.4 (0.442)	0.877 (0.877)	81.08 (0.516)
BioClinicalBert	0.822 (0.446)	0.889 (0.521)	80.4 (0.594)

Table 4: Test accuracy (and test loss) of the implementation of Bised and Mo., 2020 in comparison to our trained model

Although we used a part of the dataset, and not the whole dataset, our model's accuracy was comparable enough to the benchmark results.

Comparison with GLUE Dataset

We will compare the results of implementation of BERT for classification task with GLUE (Gluebenchmark.com, 2019) benchmark results on SST2 (Stanford Sentiment Treebook v2) with our implementation on Drug Reviews dataset. Though the datasets are different, this measure of comparison can only give a general overview of the model's performance. (Munikar, Shakya and Shrestha, 2019)

GLUE: Bert base on SST2: 91.2%

Our implementation: Bert base cased on Drug Review Dataset: 80.59%

8. Conclusion and Future Work

Our fine-tuned BERT model performed well to predict the sentiment of the drug reviews. The task of predicting the sentiments of the reviews has achieved significant levels and obtained good accuracy using the available models.

To our knowledge, a limited amount of work has been done on generating reviews (sentence prediction) using the attributes present such as rating, name, number of purchases, date. Possible future improvements could be generating drug reviews using the column attributes present in the dataset by feeding the attributes as embeddings into the neural network.

Another interesting future work, using BERT, could be Aspect-based sentiment prediction which is composed of identifying both sentiment and aspect of a text.

References

1. Korkontzelos, I., Nikfarjam, A., Shardlow, M., Sarker, A., Ananiadou, S. and Gonzalez, G.H. (2016). Analysis of the effect of sentiment analysis on extracting adverse drug reactions from tweets and forum posts. *Journal of Biomedical Informatics*, 62, pp.148–158.
2. Gräßer, F., Kallumadi, S., Malberg, H. and Zaunseder, S. (2018). Aspect-Based Sentiment Analysis of Drug Reviews Applying Cross-Domain and Cross-Data Learning. *Proceedings of the 2018 International Conference on Digital Health*.
3. Devlin, J., Chang, M.-W., Lee, K., Google, K. and Language, A. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. [online] Available at: https://arxiv.org/pdf/1810.04805.pdf.
4. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Brain, G., Research, G., Jones, L., Gomez, A., Kaiser, Ł. and Polosukhin, I. (2017). *Attention Is All You Need*. [online] Available at: https://arxiv.org/pdf/1706.03762.pdf.
5. Loshchilov, I. and Hutter, F. (2019). *DECOUPLED WEIGHT DECAY REGULARIZATION*. [online] Available at: https://arxiv.org/pdf/1711.05101.pdf [Accessed 24 Aug. 2020].
6. Alsentzer, E., Murphy, J., Boag, W., Weng, W.-H., Jin, D., Naumann, T. and Mcdermott, M. (2019). *Publicly Available Clinical BERT Embeddings*. [online] Available at: https://arxiv.org/pdf/1904.03323.pdf [Accessed 13 Aug. 2020].
7. Biseda, B. and Mo, K. (2020). *Enhancing Pharmacovigilance with Drug Reviews and Social Media*. [online] Available at: https://arxiv.org/ftp/arxiv/papers/2004/2004.08731.pdf [Accessed 23 Aug. 2020].
8. Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C.H. and Kang, J. (2019). BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*. [online] Available at: https://academic.oup.com/bioinformatics/advance-article/doi/10.1093/bioinformatics/btz682/5566506 [Accessed 19 Sep. 2019]
9. Munikar, M., Shakya, S. and Shrestha, A. (2019). *Fine-grained Sentiment Classification using BERT*. [online] Available at: https://arxiv.org/pdf/1910.03474v1.pdf [Accessed 26 Aug. 2020]
10. Gluebenchmark.com. (2019). *GLUE Benchmark*. [online] Available at: https://gluebenchmark.com/.